

Nº 8 - OCTUBRE 1984

315 PTAS.



POR FIN!

Ha llegado un Nº 1 en USA para su

Commodore 64

Ud. está en su posición vigilando Washington D.C.

La ciudad parece apacible desde su puesto de defensa y no hay los signos de la invasión Extraterrestre que Ud. espera.

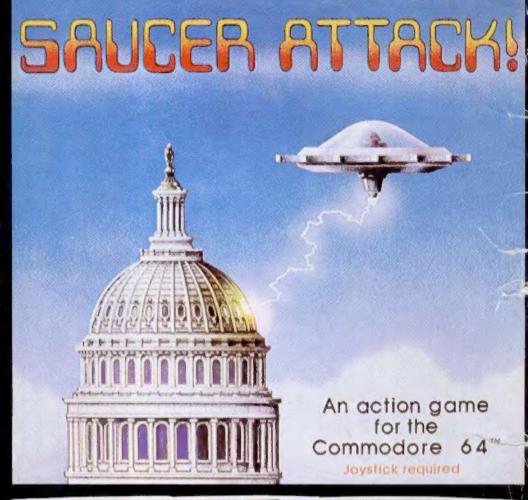
De pronto el silencio es alterado por el penetrante silbido de los platillos volantes. Son los invasores descendiendo sobre la ciudad.

Ud. entra en acción y pone en su punto de mira al platillo volante. Aprieta el botón de juego y sale un

Aprieta el botón de juego y sale un misil que destruye al ovni desintegrándose el platillo. Otro platillo volante se acerca a los edificios y antes de que Ud. se de cuenta destruye el monumento de Washington C.D. con un rayo de energía atómica. La batalla transcurre durante el día y en un momento determinado pasa a ser de noche.

Si un platillo aterriza todo está perdido. Cuando Ud. consiga neutralizar el ataque el juego continuará y se deberá destruir la nave madre. Si lo consigue una gran celebración le espera cuando vuelva a Washington D.C.

SAUCER ATTACK
es realmente
un JUEGO DISTINTO





ONADAS DIDECTANE

TOMADAS DIRECTAMENTE DE UN MONITOR 1702

COMMODORE



INSUPERABLE CALIDAD AUDIO-VISUAL

PEDIDOS

FERRE-MORET J.A.

Tuset n.º 8, entlo. 2.º Tel. 218 02 93 BARCELONA 08006

DISTRIBUIDORES
Y PARTICULARES

N	O	M	B	H	E	*
-	LE	-	0	-	10	1

DIRECCION.

.... PROVINCIA

☐ Adjunto cheque

☐ Contra reembolso

Gastos envio: 300 ptas.

Precio Venta 9.975 ptas.

SUPER-INTERESANTISIMO

1.44 CONCURSO NACIONAL ESCOLAR DE PROGRAMACION (bases pág. 5, premios pág. 13)

Ya tenemos la relación de premios que se otorgarán a los futuros ganadores del Ler CONCURSO NACIONAL ESCOLAR DE PROGRAMACION

NIVEL EGG.-100.000 pesetas en material a elegir al concursante individual o equipo. El centro representado por el ganador recibirá, a elegir, un Ordenador SX-64 ó bien 5 Ordenadores Commodore del nuevo modelo 16 con

sus correspondientes Cassettes,

NIVELES F.P. y BUP .-- 150,000 pesetas en material a elegir para cada uno de los ganadores (individual o equipo) de estas dos categorias. Los Centros representados por los ganadores recibirán, a elegir, un Ordenador Profesional Commodore con sus unidades de disco o bien 5 Commodore 16 con sus correspondientes Cassettes. Todos los ganadores, incluidos los centros, recibirán asimismo suscripciones a Commodore World y Micro-Sistemas.

Ya sólo queda el animaros a todos, centros y alumnos a darle al teclado, a la imaginación y a

REPETICION DE PROGRAMAS DE MUESTRO WE 7

Parece ser que en un momento dado, parte de la tirada de nuestro número 7, correspondiente a septiembre salió con varios listados defectuosos debido a algún fallo mecánico de la imprenta. En páginas centrales de este número aparece una separata extra con repetición de estos listados, ya que parece haber sido bastante alto el número de lectores afectados. Lamentamos este incidente, completamente ajeno a nosotros, y damos las gracias a Ibergráficas por haberse ofrecido inmediatamente a la impresión de esta separata.

HUMERO ESPECIAL DE NAVIDAD

Estamos preparando un volumen especial de programas (de 75 a 100) cara a las fiestas de Navidad. Este número especial, que esperamos poder presentar en SIMO, lo podrán adquirir nuestros suscriptores a precio de coste y el resto de nuestros lectores con un descuento del 25%. El precio de venta al público será de 1.250 pesetas. Por lo tanto el precio de nuestros suscriptores será de 625 pesetas y a los lectores no suscriptores de 935 pesetas.

La tirada tendrá un número limitado por lo que ya podéis reservar la copia o copias que desecis (boletín pie de página) pues no habra reimpresión una vez que se agoten:

EL EQUIPO AUMENTA

Diego ya tiene ayudante, Alvaro Ibáñez, que a sus 16 años y estudiante de COU es todo un joven genio del Computador. Bienvenido en nombre de todos (¡y especialmente de Diego!).

SUSCRIPTORES Y SERAS CORRECTAS

Tras haberse metido todas las suscripciones en el ordenador, si hay algún lector que observe alguna incorrección en su nombre o dirección, le rogamos nos lo comunique, así como cualquier cambio de domicilio que se

haya producido, a fin de evitar retrasos, perdidas o devoluciones.

Y hablando de suscriptores... a por el 8.000... y... después del descanso veraniego, ya nos están llegando otra vez a toda marcha.

Estamos en el suscriptor 5.670. Cuando lleguemos al 8,000 se sortearan los viajes que se indican en la página 5 (Concursos Permanentes).

Y HABLANDO DE VIAJES

Sobre el viaje a Londres nada podemos deciros de momento ya que nuestro ganador, todavia no ha tenido esa semana libre para poder ir. Sin embargo, Diego y Joan se largaron a Estados Unidos, como estaba previsto, y parece ser que no lo pasaron nada mal. Lo que Diego nos cuenta sobre la Fábrica de Circuitos Integrados de Commodore es fascinante. Pero mejor deio que él haga un resumen de sus experiencias.

PASEANDO POR "LAS AMERICAS"

El día 19 de julio Joan Pujiula i Junca llegó a la estación de Chamartín en Madrid, le dejamos refrescarse, y le enseñamos las oficinas y luego toda la capital en un rápido tour en coche.

El día 20 por la mañana temprano, saliamos de Madrid con destino a Nueva York. ¿Qué aventuras viajeras nos esperaban?

Nos recogió Doris Moris, del Instituto de Asuntos Culturales (ICA), que nos invitó a una cena-merienda-almuerzo (según el reloj con que se mire) y nos enseño parte de la ciudad. El día siguiente tuvimos libertad para reco-

rrer tiendas, ver parte de la ciudad y descansar, aunque el tiempo se puso en contra, lluvia torrencial toda la mañana, no nos logró desanimar, compramos un paraguas de los que venden a 38 en todas las esquinas de Nueva York los dias de lluvia y seguimos.

Esa tarde amainó el temporal y fuimos con otros dos jóvenes de ICA a ver el espectáculo del Radio City Hall, fue fabuloso, tanto de colorido visual como musical. La sonorización de la sala y los medios materiales (elevadores, escenarios, proyectores, etc.), lo podéis imagi-nar, era de lo mejor, para eso se trata de uno de los edificios del centro Rockefeller.

El domingo por la mañana hicimos el tour del alto y bajo Manhatan al que nos invitó Ibercharter, como podreis imaginar recorrimos todo, desde el parque de las baterías (al sur) hasta casi el Bronx (al norte), y desde Brodway-oeste y las torres gemelas al edificio de Naciones Unidas al este, es curioso, pero ese día el sol estaba tan luminoso que vendían gafas de sol en todas las esquinas de la ciudad también a 3 dólares.

El lunes por la mañana tomamos un avión desde el aeropuerto Kennedy hasta el de Fila-delfia, donde tomamos una limusine hasta el hotel Sheraton de Valley Forge, que tienen unas vistas preciosas del valle y todo tipo de instalaciones deportivas y de descanso, nosotros optamos por la segunda (lógico, ¿no?), y una cena estilo europeo (estábamos cansados

de comida americana).

El martes temprano nos recogió la limusine matrícula "CBM-1", se trata del coche de los jefazos de Commodore. Es un Cadillac larguisimo con bar, televisión, y todo dentro. Nos llevó a la planta de West Chester donde l'abrican en la actualidad una cifra escalofriante de millones de dólares en commodores 64. Creo que me bloquee la cantidad a pesetas y perdi la mitad de la frase. Nos enseñaron toda la fábrica, desde el lugar donde se recibe la materia prima al lugar donde se envian en camiones los C-64 a sus destinos, pasando por toda la cadena de producción.

Fuimos a comer con Mr. Sullivan y Mr. Kelly, que nos llevaron a dar una vuelta en la limusine y luego le dijeron al chôfer que nos llevase a la fábrica de circuitos integrados de MOS Tecnology, donde nos esperaba el señor James G. Heil (gerente de seguridad y asesor de agencias de protección de entornos en Commodore Semiconductor Systems), vimos las salas de diseño y proyecto donde se "cuecen" las ideas que luego se convierten en chips, vimos la zona de producción desde unas ventanas, nos explicaron lo que hacía cada señor y por fin pasamos a unas salas que, como las salas de descompresión en los buzos o las antecámaras de depresión de las centrales nucleares, nos preparamos para pasar a la zona aséptica, tuvimos que ponernos unas zapatillas especiales, unos gorritos para el pelo, el bigote y las cejas, y unas batas protectoras. La razón de todo esto, es que basta una pequeña particula de polvo para que un chip no sirva, igual mente durante la fabricación es suficiente una pequeña carga estática para destruir el más complejo de los circuitos integrados, y cada Waffer (base sobre la que se van depositando todas las capas que forman el circuito integrado), tarda unos 24 días en fabricarse, con lo (Pasa a la pág. 66)

BOLETIN DE RESERVA ESPECIAL NAVIDAD

NOMBRE	***************************************	EDAD
DIRECCION	POBLACI	ON
PROVINCIA	TELEFO	NO
Nº Suscriptor	ENVIO TALON	GIRO 🗀
No soy suscriptor pero adjunto	el boletín de suscripción (p	pág. 7) y me acojo
al precio especial para suscrip	tores□	
^a Precio: suscriptor 625 peseta	s. Lector no suscriptor 935	pesetas.

Convocatoria del Primer Concurso Nacional Escolar de Programación patrocinado por Commodore World y Microelectrónica y Control

BASES

- lº Podrán participar en el mismo todos aquellos miños y jóvenes que estén en edad escolar pre-universitaria o equivalente.
- 2º Habrá tres grupos definidos:
 - a) E.G.B.
 - b) B.U.P.
 - c) F.P.
- 3º Los trabajos pueden ser presentados por equipos o por individuales.
- 4º Los trabajos presentados por equipos deben especificar.
 - a) Nombre del equipo.
 - b) Centro al que representan.
 - c) Nivel de enseñanza que cursan (E.G.B.-B.U.P-F.P.).
 - d) Nombre del profesor, monitor o coordinador.
- 5º Los trabajos presentados por individuales deben ir acompañados de:
 - a) Nombre del autor.
 - b) Edad y curso que estudia.
 - c) (1) Si concursa representando a un centro: nombre del
 - mismo y nombre del profesor responsable.
 (ij) Si concursa por si mismo: nombre del colegio donde
 - estudia y firma del director o profesor responsable avalando que en tanto cuanto conoce al concursante el trabajo que presenta es suyo propio.

- 6º El tema del concurso es la realización de un programa educativo sobre materia líbre a elegír por los concursantes. La extensión del programa es también de libre elección.
- 7º Cada centro puede presentar el número de equipos o concursantes individuales que desee, siempre que cada trabajo sea enviado por separado.
- 8º Los trabajos deben incluir:
 - a) Nombre el programa.
 - b) Nombre del computador para el que está confeccionado y periféricos adicionales que se necesitan (si fuese el caso).
 - c) Objetivo del programa
 - d) Explicación detallada del programa (a máquina o impresora).
 - e) Listado completo a impresora,
- f) Grabación en cinta o disco.
 Podos los trabajos presentados deben ser originales e inéditos.
- 10° Los centros que pueden presentar equipos o concursantes individuales son;
 - a) Todo colegio, escuela o instituto, público o privado que imparta las enseñanzas de E.G.B., B.U.P. o F.P.
 - b) Centros de Informática que tengan grupos de enseñanza específica para niños y jóvenes menores de 18 años.
 - c) Clubs de informática que mantengan actividades específicas para niños y jóvenes menores de 18 años.
- 11º Los trabajos deben ser enviados antes del 31 de diciembre de 1984 a SIMSA, c/ Barquillo, 21-3º Izqda. 28004 Madrid.
- 12º El fallo del concurso se dará a conocer en la primera semana de marzo de 1985 en el lugar, fecha y hora que se indicarán.
- 13º Los trabajos premiados quedarán propiedad de los patrocinadores.

RELACION DE PREMIOS EN PAG. 13

CONCURSOS PERMANENTES

Colaboraciones de Programas y Magia

La verdad sea dicha: os estáis portando; las colaboraciones que nos están llegando son numerosas y de muy buena calidad. Por lo tanto, vamos a aclarar las condiciones de nuestros concursos permanentes.

- 1) Los sorteos se realizarán en los meses de junio y diciembre.
- 2) Se sortearán siempre seis premios, tres de 15.000, 10.000 y 5.000 pesetas en metálico de "Commodore World" y tres del mismo valor en material de Microelectrónica y Control, entre todas las colaboraciones publicadas.
- 3) Se sortearán siempre seis premios, tres de 3.000, 2.000 y 1.000 pesetas en metálico de "Commodore World" y tres del mismo valor en material de Microelectrónica y Control, estre todas las contribuciones publicadas en la sección de "Magia".
- Adicionalmente, se sortearán diversos premios de cintas, juegos, etc. donados por firmas y distribuciones, entre todas las colaboraciones.
- 5) Los autores de las colaboraciones vendidas dentro de nuestro "Servicio de Cintas" percibirán el 20% del precio de la cinta.
- 6) Todas las colaboraciones deben venir escritas a máquina y los programas grabados en cinta (si es posible), o con el listado completo en impresora. Nuestros lectores más jóvenes pueden escribir a mano pero con letra muy clara.
 - 7) Quedan automáticamente descalificadas tanto del sorteo como del

"Servicio de Cintas" las colaboraciones que hayan sido enviadas a otras revistas.

 Las colaboraciones se enviarán a "Commodore World", C/Barquillo, 21-3 izda, 28004 Madrid.

MAS VIAJES Y OTRO REGALO DE "IMPRESION"... A POR EL SUSCRIPTOR 8.000

Es evidente que a nuestros sucriptores les gusta viajar. De momento Joan Pujula visitará los Estados Unidos en julio y Juan Mejuto Iglesias irá a Inglaterra en el otoño. Queríamos sortear algo un poco especial ENTRE TODOS LOS SUCRIPTORES cuando lleguemos al suscriptor número 8.000, y hemos optado por dos premios, uno concedido por Microelectrónica y Control y otro concedido por "Commodore World".

mica y Control y otro concedido por "Commodore World".

MEC regalará una impresora MPS 801 (o similar), y nosotros hemos escogido un surtido de viajes may "chulos" para dos personas para que el ganador elija el que más le apetezca. La selección que hemos hecho es la siguiente:

Viajes de 4 días en hoteles de primera clase a uno de estos lugares: Paris, Lisboa, Atenas, Amsterdam, Niza, Viena.

O viajes de una semana en hoteles de 4/5 estrellas:

Canarias o Mallorca.

NO OLVIDEIS NUESTRA NUEVA DIRECCION: BARQUILLO, 21-3º IZDA. 28004 MADRID

TELEF.: 231 23 88

231 23 95

TIRADA DE COMMODORE WORLD Certificación de 28.000 ejemplares

Según certificado firmado por la auditoria internacional DELOITTE HASKINS SELLS, S. A. el 5 de marzo de 1984,

la tirada de Commodore World es de 28.000 ejemplares. Este certificado se encuentra en las oficinas de 5.1.M.S.A. para comprobación de cualquier persona o entidad que lo desee. Convocatoria del primer concurso nacional de programación en 80 columnas de Commodore-64 y 40-80 columnas del VIC-20 patrocinado por Ferre-Moret, S.A. Con la colaboración de Commodore-World, más de 500.000 ptas. en premios

BASES

- 1º Podrán participar todos los usuarios de COMMODORE-64 y VIC-20 que hayan obtenido la tarjeta de 80 columnas de FERRE-MORET, S.A.
- 2º Cada concursante podrá presentar, un programa realizado por él o por un equipo del que deberán indicar un titular.
- 3º Se enviará la explicación detallada del programa redactado en español, escrito a máquina. El programa se enviará en cassette o disco, adjuntando su correspondiente listado del programa.
- programa.

 4º Se enviarán por correo certificado a la siguiente dirección, FERRE-MORET, S.A. calle Buenos Aires nº 30, 2º-3.ª de BARCELONA-36. Antes del 31 de enero de 1985.
- 5º Los programas no premiados podrán ser solicitados por sus propietarios en el plazo de treinta días después del fallo.
- 6º El resultado se comunicará, directamente a los ganadores y se publicará en la revista COMMODORE WORLD.

- 7º Los premios se entregarán durante la primera semana de marzo de 1985 coincidiendo con la entrega de premios del PRIMER CONCURSO NACIONAL ESCOLAR DE PROGRAMACION.
- 8º Los programas premiados quedarán en propiedad de FERRE-MORET, S.A.
- 9º Los programas que no obtengan premio, recibirán, en caso que así lo considere FERRE-MORET, S.A., ofertas de compra de sus programas.
- 10º Los premios ascenderán a 500,000 Ptas, repartidos de la siguiente manera;
 - 1º 250.000 Ptas.
 - 2º 150.000 Ptas.
 - 3º 100.000 Ptas.
- 11º Todos los participantes, que no hayan recibido premio u oferta de compra, podrán participar en el concurso de colaboradores que habitualmente COMMODORE WORLD viene realizando con un premio extra de FERRE-MORET, S.A., de 50.000 Ptas.
- 12º Los trabajos deberán de ser inéditos y desarrollar temas de utilidad, programas profesionales, científicos, educativos, cuétera.
- FERRE-MORET, S.A. se reserva el derecho de modificar las bases de este certamen, dejarlo desierto o suprimirlo, por causa de fuerza mayor. En este caso la empresa se compromete a comunicarlo a través de COMMODORE WORLD.

Clave para interpretar los listados

Todos los listados que se publican en esta Revista han sido ejecutados en el modelo correspondiente de la gama de ordenadores COMMODORE. Para facilitar la edición de los mismos en la Revista y para mejorar su legibilidad por parte del usuario, se les ha sometido a ciertas modificaciones mediante un programa escrito especialmente para ello. Para los progra-mas destinados a los ordenadores VIC-20 y COMMODORE 64, en los que se usan frecuentemente las posibilidades gráficas del teclado, se han sustituido los simbolos grál'icos que aparecen normalmente en los listados por una serie de letras entre corchetes [] que indican la secuencia de teclas que se deben pulsar para obtener el carácter deseado. A continuación se da una tabla

para aclarar la interpretación de las indicaciones entre corchetes:

[CRSRD] = Tecla cursor hacia abajo (sin SHIFT)

[CRSRU] = Tecla cursor hacia arriba con SHIFT)

[CRSRR] = Tecla cursor a la derecha (sin SHIFT)

[CRSRL] = Tecla cursor a la izquierda (con SHIFT)

[HOME] = Tecla CLR/HOME (sin SHIFT) [CLR] = Tecla CLR/HOME (con SHIFT)

Las indicaciones [BLK] a [YEL] corresponden a la pulsación de las teclas de 1 a 8 junto a la tecla CTRL. Lo mismo sucede con [RVSON] y [RVSOF] respecto a la tecla CTRL y las teclas 9 y 10,

El resto de las indicaciones constan de la parte COMM o SHIF seguidas de una letra, número o símbolo —por ejemplo [COMM+] o [SHIFA]—. Esto indica que para obtener el gráfico necesario en el programa deben pulsarse simultáneamente las teclas COMMODORE (la que lleva el logotipo) o una de SHIFT y la tecla indicada por la letra, el número o el símbolo, en el ejemplo anterior: COMMODORE y + o SHIFT y A, respectivamente.

En los signos gráficos además se cuenta el número de veces que aparece. Por ejemplo, [7 CRSRR] equivale a 7 pulsaciones de la tecla cursor a la derecha y [3 SPC] tres pulsaciones de la barra espaciadora.

INDICE DE ANUNCIANTES

BASIC MICROORDENADORES	15	DELTABIT	50
BSP		ELECTROFICCION	51
CASA DE SOFTWARE	19,53	FERRE MORET	2, 21
CENTRO DE INFORMATICA		IDEALOGIC, S.A	33
COMMODORE		INDESCOM	67
COMMODORE WORLD (Distribución)	24	KENT ELECTRONIC	27
COMMODORE WORLD (Suscripción)	31	MICROSISTEMAS	Separata
COMPUTER CENTER		REGISTER LATELY CONTINENTAL	35
CONCURSO ESCOLAR		TELE SANT JUST	37

BOLET	IN	D	E	SI	JS	CR	LIF	CI	0	N	_	C	om	mo	dore	V	Vorld
NOMBRE DIRECCION POBLACION TELEF. CIUDAD DONDI APLICACIONES	M	ARCA Y	MOD	ELO D	DEL O	RDEN	ADOR	IBUIDO	PROVIII	NCIA			U 2.	OMM N Af 1530 F ION O SO EVISTA	ODOR NO AL PTS. DIC ME D. OLO A A (ONC	PRICHA A D A RE	
					Firma.						ANUALES) SINO A PARTIC					VIDADE: CEN EN Y QUI ORDINA DE BASIC PROGRA	
	JEN	\PLA	RES	S A	TRA	\SA	DC)S D	EC	ON	M	ODO	ORE	W	ORL	D	
Com	m R	-	-				1	2 3	4 5	6	- a	jempla	ar		275 ptas precio por che		sta el nº 5 315 ptas
Peticionario Calle Población		**********								Nº	. P		TELEF.	vincia			
Título del pro Título del pro Título del pro Precio por o Precio por o	ogram ogram	a	******	prof	grem	es a	pare		en 6	omme	•	Wo			public public	ado ado	en nº [
Peticionario Calle Población * Programa eva Incluyo chec	***********						D.	P		N	ucia		. TELE	F		******	************
Para poder so marcha un Se	tisface rvicio p	r la cre	Prime	era é dem rar fo	époc anda tocop	de Cl	eptie lub Co e los	mbre ommo ejemp	dore, lares	32 - d agota que no	enero ida en os sea	todo n solid	B4) s sus n citado	úmero s.			
	1	-		*	7												
Peticionario: Calle Población			,,,,,,,,,,							1	√o		TELE	F			
													a locality				

Commodore W O R L D

Barquillo, 21, 3º Izda. Teléf.: 231 23 85/86 28004 MADRID

Sant Gervasi de Cassoles, 39-despacho 4 o 4

Teléf.: 212 73 45 / 212 88 48

08022 BARCELONA

Commodore

WORLD

Barquillo, 21, 3º Izda. Teléf.: 231 23 85/86 28004 MADRID

Sant Gervasi de Cassoles, 39-despacho 4

Teléf.: 212 73 45 / 212 88 48

08022 BARCELONA

Commodore

WORLD

Barquillo, 21, 3º Izda. Teléf.: 231 23 85/86 28004 MADRID

Sant Gervasi de Cassoles, 39-despacho 4

Teléf.: 212 73 45 / 212 88 48

08022 BARCELONA

Commodore

WORLD

Barquillo, 21, 3º Izda. Teléf.: 231 23 85/86 28004 MADRID

Sant Gervasi de Cassoles, 39-despacho 4

Teléf.: 212 73 45 / 212 88 48

08022 BARCELONA

No te deshagas de tu VIC cuando compras un C-64

Por si no tienes suficiente con un solo ordenador



Este interface, de fácil construcción, te permitirá usar el VIC-20, el C-64, los periféricos y monitor con rapidez y facilidad

n el año 1982, compré un VIC-20 y unos meses más tarde un C-64, un monitor en color, dos unidades de disco 1541 y una impresora 1525. Fue fácil conectar las unidades de disco y la impresora a cualquiera de los dos ordenadores (pero no a los dos en un solo sistema) mediante el bus de serie I/O, dado que los dispositivos están conectados entre sí en forma de cadena.

Sin embargo, me interesaba seguir utilizando el VIC-20 con todos los periféricos ya que poseía una colección de software para el VIC (especialmente juegos) bastante completa. Además, los niños se divertían mucho con el VIC y les servía para aprender. Lo que me hacía falta era un interface capaz de hacer dos cosas; comutar el bus de seric I/O de seis conexiones que llevaba los periféricos a cualquiera de los dos ordenadores, y conmutar las conexiones acústicas y de video al monitor en color.

Como no podía encontrar ningún interface hecho, lo tenía que construir yo mismo. El interface necesitaba un conmutador deslizante o rotativo de 2 circuitos, 2 posiciones para las lineas de audio y video. (En vez de esto, se podía haber utilizado un solo conmutador de ocho circuitos)

Los conmutadores los monté en una pequeña caja de 8 × 5,5 × 10 cm., del tipo

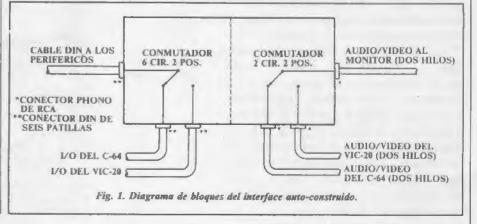
de laboratorios Multi o Retex. Instalé los conectores DIN hembra de seis patillas en el panel trasero, también monté en él dos jacks de tipo phono-RCA utilizados para las conexiones de audio y video.

Para conectar los buses serie de 1/O de cada computador a la caja interface, empleé una manguera de seis cables no blindados con conectores machos DIN de seis patillas en los extremos. Para unir los conectores de salida de audio/video, preparé cables con conectores machos de tipo DIN de 5 patillas en un extremo y jacks-RCA machos en el otro. La unión del monitor a la caja la hice con un juego

de cables de audio normales, y con otro cable de seis conductores conecté la caja del interface con el primer drive 1541 de la cadena de periféricos.

De este modo simple y sencillo, por menos de 3.000 pts., la pequeña cajainterface permite cambiar instantáneamente de un computador al otro, utilizando periféricos y monitor compartidos, sin necesidad de enchufar y desenchufar cables.

La única precaución a tener en cuenta, es apagar todo el equipo cuando se va a cambiar de ordenador, para evitar que voltajes transitorios de conmutación puedan dañar cualquier ordenador o periférico. Por supuesto los controles de color del monitor necesitarán un pequeño reajuste al pasar de un ordenador al otro. La figura I muestra el diagrama de bloques de las conexiones del interface descrito.



Viaje al centro de tu Commodore

Philip I. NELSON (RUN EEUU)

Traducido por Valerie SHANKS y adaptado por Alvaro IBAÑEZ

Vente de viaje con nosotros para explorar los misterios de la memoria de tu ordenador

¿Alguna vez has deseado poder ver lo que ocurre dentro de tu ordenador? El programa "Fisgon" convierte tu pantalla en una ventana móvil que puedes utilizar para emprender un viaje gráfico de exploración por la memoria entera de tu Commodore.

Podrás ver cómo se almacena y se organiza un programa en Basic, investigar el sistema operativo y de lenguaje Basic en la ROM del ordenador y viajar por muchos otros puntos de interés. Hablaremos principalmente del C-64, pero el dueño del VIC-20 también encontrará mucho que puede ser aplicado a su máquina, dado que los dos ordenadores se basan en el microprocesador 6502.

Para que el programa "Fisgon" funcione correctamente, tecléalo tal y como aparece en el listado, sin añadir ningún espacio ni quitar ningún REM. El programa contiene lenguaje máquina (ML), y por lo tanto es imprescindible salvar una copia antes de ejecutarlo, por si te has equivocado al teclearlo. A continuación, apaga el ordenador y vuelve a encenderlo para borrar la memoria.

carga el programa y ejecútalo.

Lo primero que aparece en pantalla son las instrucciones. Pulsa la tecla A para comprobar la dirección de la ventana de la memoria. Si los números no son iguales que los de las instrucciones, pulsa la Q para salir, y comprueba que no te has equivocado al teclear. Si los números son iguales, pulsa cualquier tecla para borrar las direcciones de la pantalla y apúntate al viaje.

10/Commodore World Octubre 1984_

si pulsas la tecla del cursor hacia abajo aparecen las direcciones de la parte inferior de la memoria, una linea de pantalla (40 bytes) a la vez. Deja pasar unas cuantas lineas a partir de la dirección 12288, donde empiezan las instrucciones; la pantalla empezará a llenarse de gráficos que no tienen ningún sentido.

Cada carácter gráfico en la pantalla representa una sola célula (dirección) de memoria en el ordenador. Existen 65.536 células en total, numeradas desde el cero hasta el 65.535 y cada célula siempre contiene algún número dentro del rango de cero hasta 255.

Puedes descubrir el número almacenado en cualquier dirección de pantalla simplemente consultando su carácter en la tabla de códigos de pantalla en el manual del usuario. Por ejemplo, el carácter * localizado a la izquierda superior de las instrucciones significa que la dirección 12288 contiene un valor de 42. La 12289 está en blanco, lo que quiere decir que el valor que contiene es de 32, etc. Ten en cuenta de que el valor cero está representado por el carácter @, y los caracteres de video inverso representan los valores de 128 hasta 255.

Ya que sabes esto, podrás deducir el valor almacenado en cualquier dirección y podrás identificar las direcciones utilizando tu comprobador de direcciones y unas sencillas operaciones matemáticas. Pero no pierdas el tiempo intentando interpretar esta "basura". Es simplemente "RAM libre", un espacio de memoria no utilizado que no contiene nada que tenga sentido. Esto aparece en cantidad antes de llegar al programa en Basic, asi que hay que seguir pasando líneas hasta que el comprobador de direcciones indique que has llegado a las direcciones 2084-3047. Pulsa la tecla del cursor si te pasas de largo.

¿Ya has encontrado el sitio? Si no encuentras las palabras en español pulsa las teclas Commodore y Shift simultáneamente. En la primera linea aparece la palabra Fisgón, con asteriscos al lado. Esta constituye la primera linea del programa en Basic, que contiene un REM. Hasta alli todo parece bastante familiar, y aparecen más palabras en español en otros sitios, pero ¿qué ha pasado con el resto del programa?

No te preocupes, las líneas que has tecleado siguen estando allí. Lo que pasa es que estás estudiando la versión "sólo para ordenadores". Para ahorrar tiempo y espacio, el ordenador abrevia las líneas en Basic a medida que las va almacenando en la RAM, comprimiendo cada palabra clave en Basic para que quede como un símbolo especial de un carácter llamado "token".

Compara esta versión abreviada con la línea 2 del listado, y verás que el comando Poke se abrevia como una W invertida. El "token" para GOSUB es una M invertida, etc.

Para que te des cuenta de la cantidad de espacio que se ahorra, piensa que este



bloque de 1.000 bytes contiene el programa "Fisgón" hasta casi el final de la línea 29. Si estas mismas líneas se presentaran en forma de listado, ocuparían alrededor de 1.650 bytes, casi las dos terceras partes más del espacio de memoria. Y esto es sólo parte del programa.

Si abrevias el texto del programa también ahorras espacio, dado que el ordenador es capaz de reconocer el "token" de un byte y seguir avanzando, en vez de tener que tragarse los tres o cuatro bytes de una palabra en Basic. El ordenador resulta más breve todavía al ser capaz de unir el final de una línea con el principio de otra. El ordenador marca el final de una línea para su propia referencia con un cero (busca el carácter @). También se abrevian los números de línea del programa, pero hay otras cosas que no pueden ser reducidas.

Como ya has visto, los REM (como en la linea 1) tienen que ser almacenados en la memoria de la misma forma que aparecen en el listado. De esta forma, puedes leer tus comentarios (REM) en las mismas "entrañas" del ordenador. Ocurre lo mismo con los nombres de las variables y los "arrays" en Basic y con cualquier carácter que metes entre comillas, como una sentencia print, por ejemplo. También quedan los espacios, con la excepción de los que aparecen después de un número de linea. Este almacenamiento literal de ciertas partes del texto normalmente no crea problemas para el potente C-64, pero el uso innecesario de los REM con el VIC-20 constituyer una pérdida de memoria que puede resultar cara.

Almacenamiento de variables

Esta zona de almacenamiento de texto es estática, y la podrías estar mirando todo el día, pero aquí no cambiaria nada. Tu programa en Basic tiene que ser almacenado como un texto fijo que el ordenador leerá mientras lo está ejecutando Como un estudiante diligente, comienza

desde el principio y va por orden, ejecutando las instrucciones a medida que avanza.

El almacenamiento de un texto en Basic en el C-64 normalmente empieza en la dirección 2048, de modo que el ordenador no tiene ningún problema en localizar el comienzo. En realidad, la dirección 2048 siempre tiene que contener un cero, de modo que el texto (incluyendo los números de línea) realmente empieza en la dirección 2049. Pero algunos programas son más largos que otros, y el ordenador necesita saber cuándo ha llegado al final de un texto. Por eso el ordenador coloca un indicador. Cuando almacena un programa en Basic, el ordenador marca el punto en la memoria donde finaliza el texto mediante el almacenamiento de dos ceros adicionales justo después del cero que marca el final de la última línea del programa en Basic.

Ahora vuelve al principio de la memoria para ver esto. Para que fuese más fácil de encontrar, dejé unas líneas en blanco donde sabía que el programa "Fisgón" terminaria en la RAM, y puse un comentario (REM) en la última línea que dice "—Fin de programa en Basic—Principio variables—". Ten en cuenta que los espacios oscuros en blanco normalmente no están presentes. Yo los puse allí para que esta primera búsqueda resultara más fácil.

¿Has encontrado el último REM? El indicador de tres ceros aparece después de la flecha. Si avanzas esta línea hasta el principio de la pantalla y haces uso del comprobador de direcciones, puedes calcular fácilmente que el último carácter del texto en Basic ocupa la dirección 5020, seguido por el indicador de fin de texto. Pero si esto es el final, ¿quién añadió todo lo demás a tu programa, y qué significa?

El culpable es el ordenador, que necesita más RAM para ejecutar tu programa que la que ocupa el texto solo. Antes de poder realizar los comandos, el ordenador tiene que reservar tres zonas adicionales de memoria para su propio uso espacio para el almacenamiento de variables, "arrays" y "strings".

A lo mejor te has preguntado alguna vez, al hacer un programa en Basic, cómo el ordenador se acuerda de los valores de todas las variables que estás utilizando. El sistema que utiliza es sorprendentemente sencillo, y lo estás viendo ahora mismo. En cuanto empiezas a ejecutar el programa, el ordenador establece una serie de casillas en la memoria, una para cada variable, y almacena el valor actual para cada variable en su propio sitio. Este espacio de almacenamiento para las variables siempre empieza después (encima) del texto del programa en Basic.

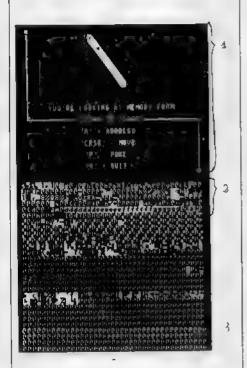
Por ejemplo, si tu programa utiliza la variable A, el ordenador reserva una casilla y le pone la etiqueta A. Cada vez que cambias el valor de A con una sentencia como A=A+1, por ejemplo, el ordenador vuelve a la casilla A para convertir su contenido en el nuevo valor. De esta forma, cada vez que utilizas la variable A en una sentencia como C=A*B, el orde-

Viaje al centro de tu Commodore

nador volverá a la casilla A para saber el valor actual de A antes de utilizaria en el cálculo

El ordenador empieza a fijar esta serie de casillas para las variables nada mas recibir el orden de ejecución, y crea un espacio de almacenamiento para cada variable en el mismo orden que las encuentra durante la ejecución del programa Para que veas cómo funciona en el programa "Fisgón", mira el listado donde aparece un GOSUB de la línea 2 a la linea 39 De esta forma se fija la rutina de lenguaje máquina y la representación de las instrucciones en la memoria

Por lo tanto, las primeras variables que encuentra el ordenador es la Z en la linea 2, seguida por la J y la Q en la linea 41 y la K en la linea 86 Fijate bien en las células de memoria después del marcador de fin de texto, y encontrarás cada uno de estos espacios para variables etiquetado con su propia letra. Estas variables no están en uso, de modo que sus casillas están inactivas. Pero las dos siguientes casillas demostrarán cómo funciona el almacenamiento de las variables



- Listo aparece nada más teclear RUN. La imagen en pantalla se desplazará hacia la derecha pero vuelve en seguida al centro. Aquí empreza el viaje por la memoria del ordenador.
- Estas son las posiciones de 0 a 255, es decir, la "página cero". Esta zona resulta muy útil para los programadores en lenguaje máquina.
- 3. El sistema operativo de la memoria también tiene que disponer de una zona de trabajo.



El programa "Fisgón" utiliza dos variables, LO y HI, para saber dónde se encuentra en cada momento ¿Has encontrado LO y HI? Ahora avanza la memoria en pantalla hacia arriba o hacia abajo, línea por línea. A medida que va pasando, aparecen nuevos caracteres justo al lado de las etiquetas de las variables, dado que el ordenador almacena los nuevos valores de LO y HI en sus casillas respectivas

Detràs de LO y HI se encuentra un espacio para la única variable de "strings", A\$. Esta casilla es la más activa. Si no la has encontrado todavía, manten pulsada la barra de espacio, y observa mientras el ordenador introduce los nuevos valores a toda velocidad en la casilla de la variable A\$. Ocurre así porque la linea 4 del programa contiene la sentencia GET A\$. y pasa una y otra vez por dicha linea

Cada vez que pasa, el ordenador almacena un nuevo valor para A\$ si se pulsa una tecla distinta. (Como verás más adelante, otra cosa está ocurriendo en la parte superior de la memoria). Las siguientes dos variables, A y B, no se actualizan hasta que utilizas el comprobador de direcciones. Verás que los valores se modifican dentro de sus casillas si tu compruebas tu dirección cada vez que se avanza una línea hacia arriba o hacia abaio

Cuando el programa llega por primera vez a la rutina Poke, el ordenador reservará un espacio para las últimas dos variables, C y D. Resulta divertido observar como los caracteres se modifican dentro de estas casillas, pero no te molestes en intentar descifrar su significado, a no ser que entiendas sistemas de numeración tan complicados como "Código EBCDC" Acepta de buena fe que tu ordenador sabe lo que está haciendo

Dado que cada programa Basic termina en un sitio distinto, y el espacio para el almacenamiento de variables se agrega al final del texto en Basic, la dirección donde comienza el almacenamiento de variables dependerá de la longitud de tu programa El tamaño de la zona de almacenamiento de variables dependera del número de variables que utilizas

Si necesitas ahorrar espacio, resulta más económico utilizar las mismas variables "corrientes" una y otra vez en una serie de bucles For... Next, como en las lineas 41 a 87 del programa "Fisgón". De no ser así, se reservará memoria para una parte de variables que sólo realizan una tarea al principio del programa y que no hacen nada más el resto del tiempo

Almacenamiento de "arrays"

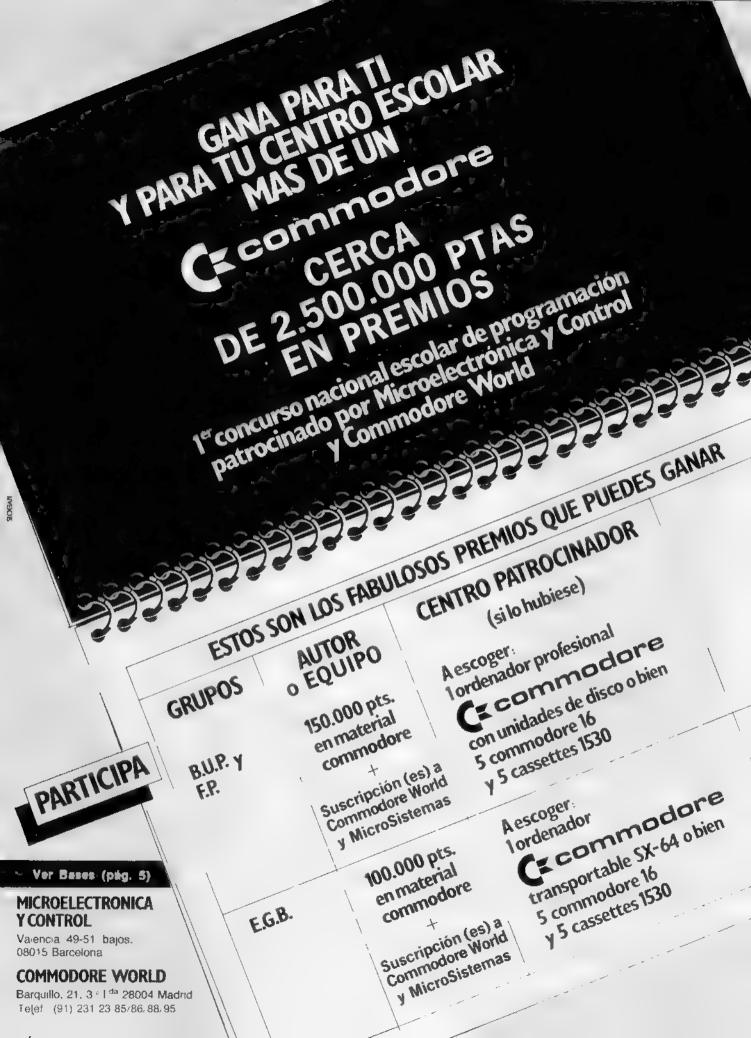
Justo encima de la zona para el alma cenamiento de variables, se encuentra otra zona especial, el espacio para el almacenamiento de "arrays". Un "array" funciona como una tabla o lista de datos dentro de tu programa. En primer lugar introduces todos los valores, y los sacas siempre que los necesitas, simplemente haciendo referencia a su posición en el "array"

El programa "Fisgón" no necesita hacer esto, pero para demostrar cómo se hace, incluye un "array" que no hace nada y que se llama ARRAY. Cada vez que pulsas la tecla Z, el ordenador asignará un nuevo valor (al azar) ai siguiente espacio (elemento) en la tabla ARRAY Para ver cómo ocurre esto, tienes que encontrar la cabecera (header) que pone la etiqueta y define el "array". Primero encontrarás los dos primeros caracteres del propio nombre, AR. A continuación siguen las cinco posiciones que definen (en formato comprimido) el tamaño y el tipo del "array".

Este "array" es de "coma flotante", de modo que el ordenador ha reservado cinco células para almacenar el valor para cada elemento. El espacio se guarda llenando las células de ceros. A medida que vayas definiendo cada elemento pulsando la tecla Z, el ordenador va metiendo los nuevos valores en estos espacios. Los valores son todos números enteros, que pueden ser almacenados en dos células, de modo que tres de las cinco células en cada espacio quedan sin usar.

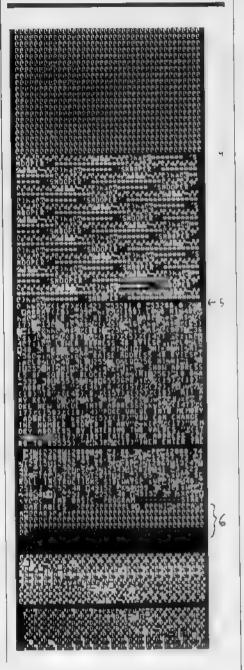
Pulsa la tecla Z unas cuantas veces, para ver cómo funciona, Recuerda que el "array" tiene 20 elementos, aunque ha sido dimensionado con la sentencia DIMARRAY (19) En este caso, como en muchos otros, el ordenador cuenta cero como un número, de modo que el primer elemento es ARRAY (0), seguido por ARRAY (1), etc

El punto que indica el fin de las variables puede modificarse a mitad de un programa si el ordenador encuentra una nueva variable por primera vez. Cuando esto ocurre, toda la zona de almacenamento de "arrays" volverá a ser "situada" en la memoria para hacer hueco para almacenar la nueva variable El ARRAY de "Fisgón" no ocupa mucho sitio en la RAM, pero ten en cuenta que el espacio para los "arrays", igual que el espacio para las variables, no se da gratuítamente.



Viaje al centro de tu Commodore

Tu ordenador no puede ejecutar un programa en Basic sin haber reservado antes la memoria necesaria para estas zonas adicionales, lo que significa que el programa ocupa más memoria después de



- 4. La memoria de pantalla. Todo lo que se encuentre en estas posiciones (1024-2039 en el C-4, 7681-8185 en el VIC-20) sale en la pantalla
- Esta es la parte del programa escrita en Basic (realmente es la segunda parte). El ordenador lo ve de esta forma
- 6. A continuación de la zona de Almacenamiento del programa en Basic se encuentra el espacio reservado por el ordenador para el registro de las variables.



ejecutarse que cuando se carga. Esto puede ser verificado con la función FRE ¿Qué es lo que se aprende de esto? Si solamente el texto ocupa casi toda la RAM, tu programa podría ser salvado y cargado sin problemas, pero podría dar un error de Out Of Memory (Memoria Agotada) en cuanto intentes ejecutarlo

Seguramente querrás saber la forma en que el ordenador se entera de la posicion de las variables y los "arrays", dado que se encuentran en una zona distinta para cada programa. No lo hace mediante espejos, sino con "punteros". De la misma forma en que utiliza una sola casilla para almacenar el valor de cada variable, el ordenador usa otras casillas para almacenar direcciones (posiciones) de los límites de estas zonas. Dadas ciertas "rarezas" en el sistema (que explicaré más tarde), se necesitan dos células de memoria conjuntas para almacenar estas direcciones

En una parte inferior de la memoria, en las posiciones 43 y 44, el ordenador alma-

cena la dirección apuntando al principio el texto del programa en Basic. Para que resulte más fácil, al encenderse el ordenador, se coloca el 2049 automáticamente en esta casilla Cuando le dices al ordenador que ejecute un programa, lo primero que hace es ir directamente a esta casilla para enterarse de la posición donde empieza el programa, y luego lo empieza a elecutar

lusto al lado, en las posiciones 45 y 46, se encuentra la casilla donde el ordenador almacena la dirección del comienzo del almacenamiento de las variables. El ordenador no puede almacenar esta dirección hasta que no termine de cargar el programa y sepa dónde termina el texto en Basic El almacenamiento de "arrays" siempre empieza al principio del almacenamiento de variables, para que el ordenador pueda usar solamente un puntero (en las células 47 y 48 de la memoria) para marcar el punto donde termina una zona y comienza la siguiente.

El ordenador mira las posiciones 49 y

algunos de nuestros programas para los ordenadores personales























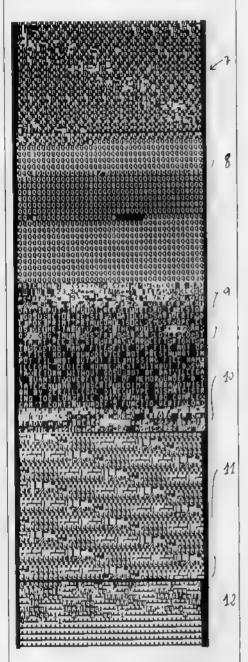




> AVD. CESAR AUGUSTO,72 Tlfs.235682 y 226544 50003 ZARAGOZA

Viaje al centro de tu Commodore

50 para saber dónde termina el almacenamiento de los "arrays". No es necesario que comprendas estos punteros para poder realizar una programación simple



- Una zona a veces vacia, a veces llena de "basura" o de RAM, una especie de tierra de nadie.
- Zona de almacenamiento de las variables de "strings". La letra Q indica que un cursor hacia arriba o hacía abajo ha sido añadido al "string" A\$
- 9. Tabla de comandos en Basic.
- 10. Aquí se encuentran los mensajes de error.
- 11. Area de control del Interface de Video VIC-II.
- 12. Sonido (registros del SID)



en Basic, ya que el ordenador lo hace él solo. Pero podrías disponer de unos cuantos trucos si sabes cómo funcionan

Hosta la página coro

De momento, vamos a olvidarnos del texto en Basic y seguir viajando hacia la parte inferior de la memoria. Justo debajo de tu texto se encuentran ocho posiciones (2040-2047) para los punteros de los gráficos sprite. Aquí no hay nada interesante, dado que m no utiliza sprites. Sin embargo, la zona justo debajo de la 2024 se está usando mucho, y si te fijas bien en esta zona verás que la pantalla empieza a hacer unas cosas muy raras

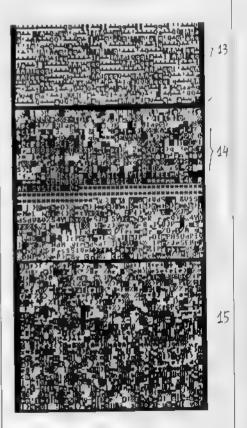
Lo que has encontrado ha sido el bloque de memoria de 1.000 bytes reservado para la memoria de pantalla (1024-2023) El ordenador utiliza esta zona para representar mensajes, gráficos, o lo que sea en la pantalla de tu televisión. Bajo circunstancias normales, si introduces cualquier valor de cero hasta 255 en cualquiera de estas 1.000 posiciones mediante un poke, aparecerá un carácter en pantalla

Pero no intentes interpretar los caracteres que ves flotando por la memoria de pantalla ahora —no tienen sentido— "Fisgón" está copiando y volviendo a copiar las zonas más altas de la memoria de pantalla en las zonas más bajas a medida que tú vayas cambiando las direcciones. Juega un poco si quieres. Verás que si cambias las líneas una por una, se crean unos efectos muy diferentes de los que ves cuando esta zona se va pasando normalmente. La pantalla no volverá a comportarse de una forma normal hasta que no hayas pasado completamente por debajo de la memoria de pantalla.

Te evitará problemas si no te olvidas nunca que la memoria de pantalla y el texto en Basic viven puerta con puerta en el C-64. Si escribes un juego de gráficos donde tienes que introducir los caracteres en la pantalla mediante unos pokes, pero te descuidas y los caracteres de tu juego pasan más allá de la posición 2023 y entran en las posiciones por encima de la 2047, harás uns pokes en el texto en Basic y tus instrucciones serán sustituidas por "basura". Para que esto no te ocurra, tienes que tener cuidado de que no te salga ningún poke más allá de las posiciones 1024-2023, o tomar otras medidas para proteger tu texto

Ahora estás muy cerca del final, y tu destino es la famosa "pagina cero" Vamos a pasar volando por las posiciones 256-1022, donde sólo se encuentra el espacio de trabajo y cosas raras para el propio ordenador. Desplázate hacia abajo hasta que el comprobador de direcciones indique que te encuentras en las posiciones 8-1007. No te preocupes si por descuido te desplazas "por debajo" de la posición cero. El programa envolvera toda la memoria del ordenador y añadira unos de sus bytes más altos a los más bajos

La página cero, es decir, las 256 posiciones de memoria mas bajas del ordenador, constituye una de las zonas mas importantes y útiles Resulta tan especial porque la dirección de cada posición es tan pequeña que cabe dentro de una sola célula de memoria, mientras que toda las



- 13. RAM de color.
- 14. Mensajes de encendido de Commodore.
- 15. Una imagen parcial del kernal. (Tiene una longitud de 8K).

demás direcciones ocupan dos células (hablaremos de esto más adelante).

Por este motivo, esta zona resulta ser tan importante para los programadores en lenguaje maquina -de hecho resulta tan valiosa que los fabricantes del ordenador han incluido en esta página unas zonas de almacenamiento especiales y muchas otras cosas que el ordenador necesita para sus propias operaciones... A diferencia de la zona permanente de almacenamiento de texto, siempre hay actividad

Sin embargo, esta zona es RAM, de modo que el ordenador puede usar unos pokes para introducir los nuevos valores en cualquier célula de la página cero. Para los curiosos y los que disfrutan con los experimentos, existen cuatro células (251-254) reservadas para que puedas hacer lo que quieras con ellas, "Fisgón" utiliza dos de ellas (251-252) para almacenar un puntero de "dirección original" para la



subrutina en lenguaje máquina.

Si quieres ver tu propia casilla, busca primero la línea de pantalla que empieza en la posición 248. Si haces un "scroll" hacia arriba o hacia abajo verás que cambia el carácter dentro de la posición 251 No confundas este espacio de almacenamiento con la zona más grande de 256-266 donde aparece un número decimal -esto es una pequeña zona de trabajo utilizada por el ordenador-.. Todas las direcciones aquí abajo caben en una célula, de modo que la segunda mitad de nuestra casilla, 252, ahora contiene solamente un cero.

En el próximo artículo, iremos de excursión por debajo de la página cero, hasta la parte superior de la memoria. Podrás ver el sistema operativo ROM, las zonas para el color, el sonido y el manejo de periféricos, y sinalmente terminaremos estudiando un poco los pokes, peeks y los cál-

culos del direccionamiento

- 1 REMW 'VIC FISGON' W 2 PRINTCHR\$(147)CHR\$(14) POKE36879,8 Z=0 DIMARRAY(19) GOSUB39 3 L0=0 HI=16 4 POKE251, LO. POKE252, HI. SYSS28 GETAS
- 5 IFA##CHR#(17)THENGOSUB13 IFAs=CHRs(145)THENGOSUB16 IFA\$=CHR\$(65)THENGOSLB19 8 IFA##CHR#\80)THENGGSUB25 9 IFA##CHR#(90)THENGOSUB36
- 10 IFA\$=CHR\$(81)THENPRINTCHR\$(147)CH R#(142) - END

11 00704

REM BAJA DIRECCION

13 LO=LO-22 IFLOK@THENLO=LO+256-HI=H I-1 IFHICOTHENHI=HI+256

14 RETURN

15 REM SUBE DIRECCION

16 LO=LO+22 IFLO>255THENLO=LO-256 HI #HI+1 IFHI>255THENHI#HI-256

17 RETURN

18 REM MUESTRA ADDRESS

19 A=(HI#256)+LO B=A+505 IFB>65535TH

ENB#8-65536

20 PRINT"[HOM][GRN3[RVSON]ADDRESS"A"

CCRSRLJESPCJTO"B"CWHTJ

22 PRINTCHR#(19)"[22SPC]"

23 RETURN

24 REM HACE UN POKE

25 PRINT"[CLR][YEL][RVSON][SPC]POKE[SPC] UNESPOIVALOR" PRINT"[CRSRD1EWHT][SHIFTC] AMBIACSECIELISPOJCONTENIBOCSPOJDE", PRINT' [CRSRD]UNA[SPC]DIRECCION'

26 PRINT"[CRSRD]ENTRANDOESPC1DOSESPC1 NUMEROS. PRINT"[CRSRD]ESHIFTE]_ESPC] PRIMERO(SPC)LA(SPC)POSICION"; PRINT

[CRSRLIY[SPC]ELISPC]SEGUNDO(SPC]EL" 27 PRINT"[CRSRD]VALORESPC]QUEESPC]QU TERES" PRINT" CRSRD1COLOCAR.

28 INPUT"[YEL][CRSRD][SHIFTE]NTRR[SPC] Z[SPC]VALORES",C.D POKEC.D PRINT"[WHT] ALGUNG[SP.]MAS?[SPC]/S/=81"

29 GETRS . IFAS=" "THEN29

30 IFA\$="6"THENZ8

31 PRINTCHR\$(147) RETURN

35 REM ARRAY MARCHDER

```
36 ARRAY(Z)=INT(RND(1)#255).Z=Z+1:IF
Z=20THENZ=0
37 RETURN
38 REM RUTINA C.M.
39 IFPEEK(820)=160ANDPEEK(850)=96THE
NRETURN
40 FORJ#828TO850 REHDQ PGKEJ, Q NEXT
41 DATA 160.0,177,251,153,0.30,200.2
08,248
42 DATA 230,252,160,0,177,251,153,0,
31,200
43 DATA 208,248,96
44 FORJ=5482T05635 POKEJ,32 NEXT
45 REM MOVIMIENTO LATERAL
46 FORJ=0T030 POKE251, J. POKE252, 16 8
YS828 FORK=1T018 NEXTK, J
48 RETURN
49 REM C-FIN PROGRAMA EN BASIC--INIC
10 DE VARIABLES>
```

```
1 RE20米市市市市市市 「FISBON」 中央市市市市市市市市市市市市市市
2 PRINTCHR#(147)SHR#(142) - POKE53281,
0 POKE53280.0.Z=0 DIMARRAY(19) GOSLB
39
3 PRINTCHR$(159) LO=0 HI=48
4 POKE251, LO POKE252, HI SYS49152-GET
自集
5 IFR#=CHR#(17)THENGUSUB13
  IFH$#CHR$(145)THENGOSUB16
  IFH#=CHR#(65)THENGOSUB19
8 IFA#=CHR#(80)THENGOSU825
9 IFR##CHR#(90)THENBUSUB36
10 IFR##CHR#(70)THENPRINTCHR#(147)CH
R#(142) END
11 GOT04
12 REM BAJA 40 BYTES EL FINAL DE LA
MEMORIA
13 LO=LO-40 IFLOCUTHENLO=LO+256 HI=H
1-1-IFHICOTHENHI=HI+256
14 RETURN
15 REM SUBE 40 BYTES EL FINAL DE LA
MEMORIA
16 LO=LO+40 · IFLO>2557HENLU=LO-256 · HI
=HI+1:IFHI>255THENHI=HI-256
```

```
17 RETURN
18 REM MUESTRA LA DIRECCION DEL BLOQ
JE FUENTE
19 A=(HI#256)+LO B=A+999: IFB>65595TH
ENB-85536
20 PRINTCHR#(19)CHR#(18)"DIRECCIONES
"CHR$(146), A, CHR$(157) CHR$(32)
21 PRINTCHR$(18)"[CRSRL]"CHR$(146);-
9. CHR$(157)CHR$(32)CHR$(18)"[SPC]PUL
SRESPECIUNAESPECITECLA"
22 GETAS IFAS="THEN22
23 RETURN
24 REM POKE EN UNA DIRECCION
25 PRINTCHR$(147)CHR$(17)CHR$(158)CH
R#(18)TAB(4)"POKE[SPC]EN[SPC]LA[SPC]
MEMORIA"
26 PRINTCHR#(5)CHR#(17)"PUEDES[SPC]C
AMBIHRISPOJELISPOJCONTENIDOISPOJDEISPOJ
CUALQUIER"
27 PRINTCHR#(17)"POSICION[SPC]DE[SPC]
MEMORIALSPOJINTRODUCIENDOLSPOJDOS"
28 PRINTCHR#(17)"VALORES. " . PRINT"[CRSRD]
[SPC]PRIMERO(SPC]ENTRA(SPC]LA(SPC]DI
RECCIONISPOJ(ØISPOJAISPOJ65535)
30 PRINTCHR#(17) "QUECSPCJQUIERES(SPC)
CAMBIAR. "CHR$(17)CHR$(17)CHR$(158)
31 INPUT"ENTRAISPCJ2[SPC]VALORES", C.
D POKEC, D PRINTCHR$(17)CHR$(5)"OTRO[SPC]
MASI2SPC]'S'=81[SPC]"
32 GETAS IFAS=""THEN32
33 IFR##"S"THENPRINTCHR#(158) 80T031
34 PRINTCHR$(159) RETURN
35 REM DEFINE ELEMENTO DEL ARRAY
36 ARRAY(Z)=INT(RND(1)#255):Z=Z+1:IF
Z=20THENZ=0
37 RETURN
38 REM CARGADOR BASIC PARA LA RUTINA
 DE CODIGO MAUJINA
39 IFPEEK(49152)=160THENRETURN
40 POKE214,12 PRINT: PRINTTAB(10)CHR$
 (18)CHR#(5)"[25PC]UNESPC]MOMENTO[25PC]
 41 FORJ#49152TO49200 READO POKEJ.O.N
 EXTJ
 42 DATE 160,0.177,251,153.0.4
 43 DATA 200,208,248,230,252
 44 DATA 160.0,177,251,153.0,5
 45 DATA 200,208,248,230,252
 46 DATA 160,0,177,251,153,0,6
 47 DATA 200,208,248,230,252
 48 DATA 160.0,177,251,153,0,7
 49 DATA 200,192,232,208,246,234
 50 REM LA SIGUIENTE RUTINA DE C.M.
LIMPIA LA MEMORIA DE COLOR
 51 FORJ=49201TO49220 RERDO POKEJ.O.N
 EXT
 52 DATA 162.0,169.1,157.0,216
```

```
53 DATA 157,0,217,157,0,218,157
54 DATA 8,219,232,208,241,96
55 REM DATOS DE INSTRUCCIONES EN LA
PANTALLA
56 DATA 32,160,134,137,147,135
57 DATA 143,142,160,5,19,20,1
58 DATA 19,32,22,9,5,14,4,15
59 DATA 32,12,1,32,13,5,13,15
60 DATA 18,9,1,32,4,5,19,4
61 DATA 5,49,50,50,56,56,32,32
62 DATA 1,32,49,51,50,56,55,39
63 DATA 1,39,32,61,32,1,4,4,18
64 DATA 5,19,19,32,40,4,9,18,5,3,3,9
, 15, 14, 41
65 DATA 39,3,18,19,18
66 DATA 39.32.61.32.13.15.22.5.18
67 DATA 39,16,39,32,61,32,16
68 DATA 15,11,5,39,6,39,32
69 DATA 61,32,6,9,14,32
70 REM ESCRIBE LAS INSTRUCCIONES
71 FORJ=12288T013327 POKEJ 32 NEXT F
ORJ=5128T05327 POKEJ 32 NEXT
72 FORJ=12342T012678STEP41 READQ POK
EJ, Q NEXT
73 FORJ=12813T012841 READQ POKEJ,Q.N
EXT
74 FORJ=12900T012913-READQ POKEJ,Q N
EXT
75 FORJ=12980T013004 · RERDQ · POKEJ · Q · N
EXT
76 FORJ=13060T013073 READG POKEJ Q:N
EIIT
77 FORJ=13141T013150 READQ POKEJ Q N
EXT
78 FORJ=13221T013230 READQ POKEJ Q N
EXT
79 POKE12288,42 POKE13287,42 FORJ=12
328T012880STEP40 . POKEJ . 66 NEXT
80 POKE12888,74 FORJ#12889TD12898 PO
KEJ, 67 - NEXT
81 POKE12927,73 FORJ=12915T012926 PO
KEJ, 67 - NEXT
82 FORJ=12967T013247STEP40 POKEJ,66
NEXT
83 FORJ=12343T012638STEP41:POKEJ,223
 NEX1
94 FORJ=12382T0126778TEP41 PCKEJ.95
NEXT
95 REM SCROLL LATERAL DE INSTRUCCION
86 FORJ=0T040 POKE251, J POKE252, 48 S
Y849152 NEXTJ
87 FORJ=40T00STEP-1 . POKE251 . J . POKE25
2,48 SYS49152 NEXT
88 RETURN
89 REMC--FIN DE PROGRAMA BASIC----PR
 INCIPIO VARIABLES>
```

OFERTA ESPECIAL

(limitada)

Commodore C-64

en 48 horas a sólo 54.990 ptas. Entregamos en su domicilio toda clase de aparatos periféricos a precios excepcionales Para Comercio
hay condiciones muy,
muy especiales...!

Consultenos!

LOBERCIO, S.A. — Info-Import — Avda. Cánovas del Castillo, 4, 4º B — MALAGA — Telf. (952) 44 82 64/21 12 91

casa de software s.a.

PARA COMMODORE 64

PRACTICALC 64



MANUAL EN CASTELLANO

PROCESADOR DE TEXTO

35 000 caracteres, 240 columnas Versión diskette: 21,500,— Versión cartucho: 24,900,— (grabación de documentos en diskette y cassette)

HOJA DE CALCULO

2 000 coordenadas (funciones matemáticas, sort, gráficos...) Versión cassette C-64: 15.500,— Versión cassette VIC-20: 11.500,— Versión diskette VIC-20: 13.500,—

CONTABILIDAD 64 PROFESIONAL



P.V.P.: VERSION A: 24,550,-VERSION B: 29,500,

VERSION: A

300 cuentas 3 000 apuntes

- Listado de diario
- Balance de situación y de Sumas y Saldos

VERSION: B

600 cuentas

2.300 apuntes

- Estractos de cuenta
- Cuenta de explotación, diario de cierre
- Mantenimiento de ficheros, utilitario...

EQUIPO NECESARIO:

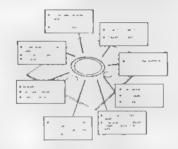
- COMMODORE 64 y unidad de disco
- IMPRESORA MPS 801 (versión A y B)
- IMPRESORAS CENTRONICS (versión B)

ADAPTADO AL PLAN GENERAL CONTABLE ESPAÑOL

PROGRAMAS EN CASSETTE

CONTABILIDAD PERSONAL	00,-
CONTABILIDAD DOMESTICA	00, -
CONTABILIDAD DOMESTICA (VIC-20)	-,00
PERT	-,000
CALCULOS FINANCIEROS	-,00
CALCULOS ESTADISTICOS	00,
FICHERO MONEDAS	-,00
FICHERO SELLOS	-,00
FICHERO BIBLIOTECA3.0	00,-
FICHERO AGENDA	00,~
QUINIELAS	00

VIZAWRITE 64



MANUAL EN CASTELLANO

PROGRAMAS EN CARTUCHO + DISKETTE

VIDEOCLUBS 64

1.800 videofilms 650 clientes por disco Saldo de cada cliente Altas y bajas videofilms Búsqueda videofilms/clientes.

P.V.P.: 35,000.-

AUMINISTRACION DE FINCAS

30 inmuebles
400 inquilinos
Listado recibos
Propiedad vertical
Comunidades de propietarios
P.V.P.: 35,000,

LAPIZ OPTICO Y SOFTWARE GRAFICO

P.V.P.: 12.500,-

JOY STICK DE PRECISION

PVP 2950,-

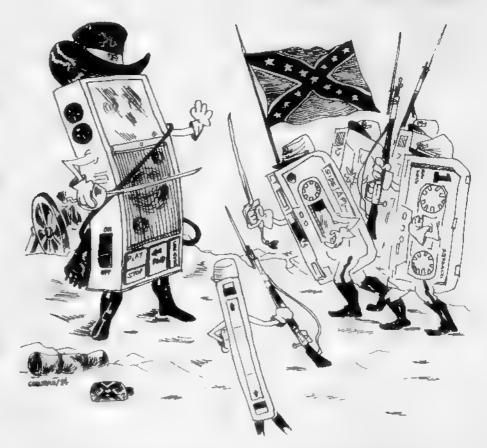
DE VENTA EN TIENDAS Y DISTRIBUIDORES ESPECIALIZADOS

casa de software, s.a. c/. aragón, 272, 8.º, 6.ª tel 215 69 52 08007 barcelona DESEO RECIBIR INFORMACION.

Nombre y dirección

COMODORE 64 ES UNA MARCA REG. DE COMMODORE BUSSINES MACHINES, INC.

Qué hacer con tu datassette para que entren esas cintas rebeldes



n alguna ocasión habréis tenido la oportunidad de intercambiar programas con alguien, pero al llegar a casa con la cinta grabada en otro datassette, ¡qué desilusión!, no hay manera de hacer entrar la cinta, unas veces da "LOAD ERROR", otras se para antes de cargar el programa, o sólo una parte y se queda "colgao en las alturas"... por cierto al datassette no le iría nada mal... un repaso a sus ajustes

Vamos a ver cuáles son las posibles causas de que las cintas no carguen en nuestros ordenadores

Uno de los principales problemas que suelen aparecer es que la cinta se frene; la solución para eso suele ser como muchos saben, darle unos golpecitos contra la mesa, ¡pero sin pasarsel, a la cinta para colocar todas las vueltas al mismo nivel, y darle varias pasadas a la cinta de un lado a otro con las teclas REWIND y FAST FORWARD, de este modo conseguiremos igualar la tensión y la posición de todas las vueltas de la cinta

El otro problema suele ser la suciedad acumulada en la cabeza, su remedio es utilizar una de las cintas limpiadoras que están comercializadas para los equipos de alta fidelidad, ya que aunque el datassette tenga salida digital, su funcionamiento es simiar al de los normales de sonido. También se podrian limpiar las cabezas de borrado y grabación/lectura con un algodón empapado en alcohol diluido, teniendo cuidado de no dejar restos de algodón enganchados en los mecanismos.

Si todo lo anterior falla, pasamos a lo más delicado por llamarlo de alguna manera, se trata del ajuste de la posición de la cabeza. Este ajuste no debe tocarse si no se trata de un caso de auténtica necesidad, pero algunas veces podemos comprar una cinta comercial que nos cuesta bastante dinero, la probamos y no funciona, la devolvemos a la tienda, nos la cambian por otra, perdemos tiempo y dinero, y finalmente nos damos cuenta de que es nuestro datassette el que no está alineado como los demás. Al igual que con cintas comerciales, nos puede suceder con cintas grabadas en otro datassette, la solución no es demasiado complicada, pero necesita realizarse con cuidado

Si tenemos el modelo antiguo de datassette (el que era más alto y alargado), la cosa se complica, ya que no permite acceder al tornillo de ajuste desde el exterior, y es necesario desmontar casi todo el datassette para acceder al ajuste. Pero con un poco de suerte tendremos el modelo nuevo (el bajito, cuadrado) En el frente junto al borde de la abertura para introducir la cinta, podemos ver un agujero circular que permite introducir un pequeño destornillador para realizar el ajuste sin tener que abrir la unidad

Si disponemos de un osciloscopio, podemos colocar la punta de prueba tocando la placa de circuito impreso del datassette a través de un agujero que podemos localizar en la parte inferior de la unidad, a través de él tocaremos la pista que conduce la señal de lectura de la cabeza lectora/grabadora, esta señal es emuy pequeña y deberemos ajustar la sensibilidad del canal vertical a unos 20 mV./cm. (la señal debe ser de unos 80 mV

de pico a pico cuando tengamos una cinta grabada colocada en la unidad y el pulsador play presionado). El ajuste deberá realizarse moviendo el tornillo hasta conseguir la máxima señal en el osciloscopio.

Aquellos que no dispongan de instrumental, deberán realizar el ajuste por el "metodo experimental", girando el tornillo media vuelta en un sentido, probando si la cinta que deseamos cargar entra bien o la cosa va peor, si va mejor pero no del todo giramos otro poco, y así hasta conseguirlo. Si al girar fuese peor que al principio, deberemos girar el tornillo en el otro sentido y probar de nuevo.

NOTA IMPORTANTE

Es necesario anotar los movimientos del tornillo que vamos realizando para luego poder dejarlo en el mismo punto que estaba, ya que si lo dejamos muy cambiado no entrarán las cintas que fueron grabadas por nuestro datassette aunque lo hagan correctamente las otras.

Algunas veces es posible dejar el tornillo en un lugar en el que cargan todas las cintas, si fuera posible, ese sería el ajuste correcto, ya que podríamos recuperar los programas que antes teníamos y los nuevos (comerciales o prestados).

Finalmente conviene volver a sellar ese tornillo con una gota de laca de uñas, de este modo no se moverá el tornillo de ajuste.

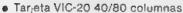
Espero que estos consejos ayuden a resolver los problemas con los intercambios de cintas y la adquisición de programas comerciales.

INCREIBLES TARJETAS PARA COMMODORES 64 Y VIC-20 **DE 80 COLUMNAS!**

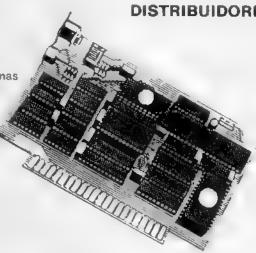
LA TARJETA DE 64K ES LA PERFECTA COMBINACION PARA LA TARJETA DE 40/80 COLUMNAS:

- Puede escoger entre 22/40/80 columnas y obtiene 31743 bytes libres!!!
- E sofware del EPROM de la tarjeta le permite utilizar todos los comandos de ficheros (OPEN CLOSE PRINT, INPUT GET SAVE and LOAD) con los 31232 Bytes de Memoria «oculta», lo que produce la sensación de qué hay 8 cassettes super rápidos conectados a su Vic-20 Imagínese... menos de 1 segundo para guardar 28 K™





- Tarieta COMMODORE 64 80 columnas
- Tarjeta VIC-20 Ampliación de memoria 64 K - RAM
- Siots de expansión para VIC-20 y 64 de 2 y 5 para conexión de tarjetas



PEDIDOS

FERRE-MORET ZA.

DISTRIBUIDORES Y PARTICULARES

C/Tusset, 8 - ENTRESUELO 2ª Teléf.: 218 02 93 08006 BARCELONA

ATENCION!

SOFTWARE 80 COLUMNAS

Estos programas requieren las tarietas distribuidas por FERRE-MORET S. A. Para el VIC-20, aunque los programas funcionan con un minimo de 16K, están diseñados para su ut ización con la Tarjeta de 64K RAM Capacidad de almacena,e il mitada. Todos los programas se venden con un manua, expicativo en castellano

AGENDA DOMESTICA Y DE NEGOCIOS (REF. ADD4080C)

noluye una función de busqueda que identifica la totalidad o parte de los nombres, direcciones o numeros de teléfono, es decir, el sistema permite identificar un nombre dirección o teléfono, aunque unicamente se recuerde parte de la información. Si el usuario no recuerda una dirección concreta lamando el nombre de la población, obtendrá todas las direcciones de esa población

CONTROL STOCK 40 COL. (REF. st40C) CONTROL STOCK COL. (REF. st80C)

Posee una función de busqueda que permite visualizar productos concretos especificando el producto, su descripción, lugar de almacena je proveedor, dirección del proveedor, código de identificación del producto, precio de coste, precio de venta, cantidad (en paquetes, cajas etc., valoración del stock niveles máximos y minimos de stock. Si además se dispone de impresora, se pueden obtener listados de productos específicos y listados de todo el almacén.

PEDIDOS, ALBARANES FACTURAS (REF. IDO4080CP)

Se precisa la tar, eta de 64K RAM impresora. El programa permite imprimir encabezam ento con el nombre de la Empresa y su dirección y se puede elegir que el documento sea una Factura. Albarán o Pedido. Otros detalles del encabezam ento son inumero de factura, fecha, numero de pedido del ciente referencia del cliente, etc. El programa permite la introducción de var os ITE's especifica el producto, descripción cantidad, preciolantes de impuestos precios con impuestos incluidos, gastos de envio etc. Numera ción automática de los documentos. Permite hacer copias

DIETARIO DOMESTICO

Y DE NEGOCIOS (REF. HBDBOCDP)
Funciona con Cassette y Unidad de Disco El programa pide la fecha e identifica cualquier entrevista concertada para ese dia deta ando con quién es la entrevista, motivo de la misma, hora y lugar de la reunión duración de la reunión, etc

Permite la busqueda diaria o mensual para identificar reun ones por la fecha, o lugar de reunión, o persona, tanto en el pasado como en el futuro. y además visualiza un calendario

CONTROL DE ALMACEN A 80 COL. (REF. ST80CDP)

Los datos se pueden almacenar en cassette o Unidad de Disco. Se incluye la descripción de los productos, su referencia o numero, cantidad precio de compra, precio de venta, nive es máx mo y m nimo de existencias, pedidos minimos exigidos por los proveedores, nombre y dirección del proveedor, etc. Se dispone de una función de busqueda, que permite visualizar todo el almacén productos cuya cantidad exceda, a máxima autorizada, productos con existencia por debajo del nive im nimo requer do, salidas de almacén, valor del stock, precios y márgenes

FICHERO DE PERSONAL A 80 COL. (REF. PRF80CDP)

Se necesita impresora. El programa incluye el nombre dirección teléfo. no, edad fecha de nacimiento, numero sexo estado numero de Segur dad Social, Cualificaciones. Historial de empleo Estudios sueldo funciones que realiza actualmente, programas de training a los que ha as stido etc. Se pueden visualizar todos los datos o seleccionar los que se desean mediante la utilización de una función de busqueda. Excelente para selección de personal, ya que permite identificación por nombre, pero también se puede buscar por una qua ificación o característica concreta por a edad, experiencia, etc.



Pasaporte al mundo (I)

Amplia tus horizontes sobre computadores con este artículo que os abrirá nuevas puertas a tí y a tu Commodore. La clave está en la puerta del usuario de tu Commodore.

Este artículo describe cómo usar tu computador Commodore en una aplicación "real". Esto incluye la conexión de tu computador a aparatos externos. En cualquiera de estas aplicaciones existe el riesgo, por pequeño que sea, de que el aparato externo dañe al computador.

Vamos a describir las técnicas de interfaz necesarias, pero si no estás acostumbrado a la electrónica, no intentes conectarlo hasta no conseguir que te ayude alguien que sepa de esto.

Tiajando por el mundo descubres cosas may interesantes. Como, por ejemplo, qué hacer si no sabes ni remotamente de qué forma puedes llegar a tu destino. Se trata de aprovechar los lugares de conexión. Desde uno de esos lugares se te abren las puertas para acceder a mil otros. Supón que, estando en un pequeño pueblo de la provincia de Barceona, quieres visitar a un amigo que vive en otro pequeño pueblo de la provincia de Madrid. A lo mejor en algún lugar puedes tomar un autobús a un pueblo del de allado, luego ir en el carro de un campesino, convencer a alguien para que te lleve un poco y andar el resto del camino hasta otro pueblo en que... y no llegarás nunca.

En cambio, nay lugares que son como centros de comunicaciones. Una vez has llegado alli, sabes que te será muy fácil llegar a tu destino. Si quieres ver pronto a tu amigo, sabes que seguramente saldrá de tu pueblo algún autobús con destino a Barcelona. Y, una vez en ese centro de comunicaciones, sabes que puedes llegar a Madrid. Y aunque nunca hayas estado en Madrid, puedes estar seguro de que encontraras un autobús que te lleve al pueblo de tu amigo. Estos centros de comunicaciones no existen sólo en los transportes, también la compañía telefónica y correos trabajan de esa forma. Incluso tu Commodore 64 o VIC-20 tienen incorporado un centro de comunicaciones, la puerta del usuario.

Da la vuesta a tu computador y obsérvalo por atrás. En el lado derecho verás un conector de doce patas y doble cara, con lo que tienes 24 conexiones en él. A través de este conector puedes contactar con el resto del mundo de los computadores, por ejempio usando un modem para que tu Commodore hable por teléfono. Además, puedes comunicarte con cualquier aparato que tenga interfaz "RS232" (una cosa muy común aunque quizás te suene nuevo), puedes controlar las luces y tu equipo de música, encender y apagar tuaire acondicionado (si tienes) o la calcfacción, tomar datos meteorológicos y hacer mil y una cosas que se te ocurran Para que puedas controlar todas estas posibilidades tendrás que enterarte un poco de cómo los Commodore leen y escriben en la puerta del usuario. Hay algunos registros importantes que hay que conocer y un poco de aritmetica de computadores ayudará bastante. Vamos a echarle un vistazo a esos registros para que puedas conseguir que hagan lo que tú quieres.

de registros

El VIC-20 y el Commodore-64 disponen de diversos registros de datos bidireccionales. Esto quiere decir que cada bit del registro se puede usar como entrada o como sal.da de datos. Incluso el joystick funciona a través de este tipo de registros.

Los registros del VIC-20 se encuentran en las posiciones de memoria 37137 para la puerta A y 37136 para la puerta B En el Commodore 64 los registros están en la posición 56576 para la puerta A y 56577 para la puerta B. Pero, antes que nada, hay que decirle al computador cómo vamos a usar cada uno de los bits de estos registros (y, con ello cada una de las líneas de datos de la puerta del usuario), si como entrada o como salida).

Para esto escribiremos un número en el registro de dirección de datos (data direction register, DDR en la literatura inglesa) que se encuentra en la posición 37139 (puerta A) y 37138 (puerta B) en el VIC-20 y 56578 (puerta A) y 56579 (puerta B) en el C-64. En este articulo vamos a usar solamente la puerta B de ambos Commodores.

(Recuerda que existe una diferencia significativa entre el C-64 y el VIC-20: el VIC-20 usa como circuito de entrada/salida una VIA 6522 y el C-64 dos CIAS 6526. Esto es importante en cuanto a la puerta de joystick, que es la puerta A en un VIC y la segunda CIA en un C-64. Por ahora, basta con que recuerdes que existe esa diferencia).

No hay una orden especial para ajustar el DDR, si no que (¡cómo no!) hay que "pokear" en el registro el valor adecuado, una vez calculado con aritmética binaria.

Calma, no te asustes, ni pases al articulo siguiente, ni te vayas a dar de comer al perro Esto no es nada difícil. Hasta tuprofe de matematicas sabria hacerlo bien. Los DDR, como todas las tripas de tucomputador, hablan en binario. Cuando ves la representación binaria de un registro, ves a la derecha el bit menos significativo (least significant bit, LSB) y a la izquierda el bit más significativo (most significant bit, MSB). El registro comprende un solo byte (-8 bits) de memoria, así que numera los ocho bits de 0 a 7, 0 a la derecha, 7 a la izquierda. Para que veas lo fácil que es todo el sistema binario, te voy a explicar cómo se parece al sistema decimal, el sistema de números normal.

Toma un número cualquiera. Por ejemplo 2546

2546-2×1 000+5×100+4×10+6×1 (¿estás de acuerdo?)

Fijate en que vamos multiplicando, de derecha a izquierda, cada citra por diez



veces lo que multiplicábamos la anterior. (1, 10, 100, 1.000...). Esto es así porque éste es el sistema de numeración del 10. Si en decimal las columnas de cifras se multiplican, de derecha a izquierda por 1, 10, 10×10, 10×10×10, 10×10×10..., en binario se multiplican por 1, 2, 2×2, 2×2×2, 2×2×2×2, 2×2×2×2×2... Por ejemplo 1011B 1×1+1×2+(1×2×2+1×2×2×2 11D) (hemos empezado esta vez por la derecha)

(B=Binario, D=Decimal)
El número en sí es el mismo, pero su representación es diferente. Es lo mismo 10118 pesetas que 11D pesetas, pero una es la forma de pensar del computador y otra es la tuya. Para conseguir que el computador "haga algo" necesitamos muchas veces meter un número en un registro y ese número es tácil saber cuál es en binario. Pero para "pokearlo" hay que dárselo al Commodore en decimal, ya sabes I ntonces sacas tu calculadora y multiplicas y sumas como hemos hecho antes

Para estos cálculos te sera útil la tabla de notencias del 2

an linetarion mat m			
1	-1	2	()
2	=2	=2 ^	1
2×2	=4	=2 ^	2
2×2×2	=8	=2 ^	3
2×2×2×2	=16	-2 ^	4
2×2×2×2×2	=32	=2 *	5
2×2×2×2×2×2	-64	=2 ^	6
2×2×2×2×2×2	=128	=2 °	7
(asi tendrás que mult	nplicar me	nos)	

El VIC-20 y el Commodore 64 disponen de diversos registros de datos bidireccionales.

Esto quiere decir que cada bit del resgistro se puede usar como entrada o como salida de datos. Incluso el joystick funciona a través de este tipo de registros.



El poli controla el bit

Una vez vistas estas reglas, ya podemos calcular los valores a introducir en el DDR para los diversos bits según queramos emitir o recibir datos a través de ellos.

Vamos a escribir un byte, esto es, ochobits. Cada bit controla una linea de entrada/salida (una pata del conector). Si el bit vale 0, esa es una entrada, y si vale I es una salida. Inicialmente, al enchufar el aparato, todas las lineas son entradas. Si es esto lo que quieres, no tendrás que escribir nada en el DDR. Si quieres tener también salidas, tendrás que escribir en el DDR un número que, puesto en binario, tenga a 1 los bits de las líneas que quieras que sean salidas y a 0 los que quieras que sean entradas. Recuerda: Una entrada → bit correspondiente a 1 en el DDR, una salida -> bit correspondiente a 0 en el DDR

Por ejemplo: De las ocho líneas (las numeramos 0... 7) queremos que sean todas entradas menos la 2 y la 4. Si ponemos una S para salida y una E para entrada, el byte del DDR será.

Nr. de bit 7 6 5 4 3 2 1 0 entrada/sahda E E E S E S E E E C O/1 0 0 0 1 0 1 0 0

Es decir, que tendremos que escribir en el DDR el número binario 000101008 (B=binario). A nuestro Commodore se lo

Una vez vistas
estas reglas, ya
podemos calcular
los valores
a introducir en el DDR
para los diversos
bits según queramos
emitir o recibir
datos a través
de ellos.

tenemos que contar en decimal, pero ya hemos aprendido a hacerlo. Cogemos los bits uno por uno, comenzando por atrás y vamos multiplicando por los números de la tabla anterior de potencias del dos.

00010100B= 0×1+1+0×2+1×4+0×8+1×16+0×32 +0×64+0×128 -4+16-20

O sea que tenemos que pokear (4+16=20) en el DDR. Como ves, al escribirlo de la forma POKE 37138,(4+16) y no como POKE 37138,20 es más fácil ver que hemos puesto a 1 los bits 2 y 4 (puedes comprobarlo con la tabla del párrafo anterior) y a 0 los demás.

El DDR, o "política de tráfico", hace que la comunicación circule en la dirección correcta. Ya sólo queda escribir en los registros de datos los bits de salida (con un "poke") y leer en los mismos

Los relés los puedes adaptar tú mismo con una etapa amplificadora de un solo transistor, si sabes electrónica; y si no, a lo mejor puedes usar una de las cajas de relés que se venden para el VIC-20 y el C-64.

registros de datos el valor de los bits de entrada (con un "peek")

Ya hemos visto como "pokear" para poner los ocho bits de un byte al valor que queramos. Ahora nos queda separar los datos de entrada entre si. El número que obtienes al hacer un peek contiene, en forma codificada, el valor de los ocho bits de datos. Si sólo deseas saber el valor de uno de estos bits, no tienes más que poner una "máscara" sobre el registro, de forma que sólo veas el bit que deseas comprobar. Esta máscara es como una plantilla que no deja ver más que uno o varios bits seleccionados de los ocho que hay en un byte.

En el lenguaje BASIC del Commodore, la máscara se "pone" con la función AND. La máscara en si es un numero: Es la potencia de dos del número de, bit a comprobar. Si, por ejemplo, quieres comprobar el bit 3, de la tabla de potencias del dos ves que el número a usar como mascara es 2^3 = 8. La orden de programación podría ser

IF PEEK (37136) AND 8 = 1 THEN GOTO 500

Esta orden provocaria un saito a la linea 500 si la linea de entrada 3 estuviera a nivel 1

Libre comunicación

Armado con estos conocimientos, ya puedes comenzar a comunicarte desde tu Commodore 64 o VIC 20 con el mundo exterior. Supón que deseas que el computador ponga en marcha una alarma cuando se abre la puerta principal y que a continuación imprima en pantalla la hora a la que sucedió. Puedes conseguir esto comprobando si está abierta la puerta, para lo que puedes usar un simple interruptor.

Con el mismo método puedes utilizar tu Commodore como temporizador de programas para la televisión o para el video. Escribe un programa que tenga en cuenta todos los acontecimientos que quieras. Después usa unos reles como salida para encender o apagar la television o el video automáticamente cuando se cumplan las condiciones que hayas establecido. Los relés los puedes adaptar tú mismo con una etapa amplificadora de un solo transistor, si sabes electronica; y si no, a lo mejor puedes usar una de las cajas de relés que se venden para el VIC-20 y el C-64 (consulta los anuncios publicados en COMMODORE WORLD).

Si te interesa medir la velocidad dei viento, la temperatura, o el nivel de líquido en un tanque, necesitas un convertidor analógico-digital. Si tu aplicación no exije demasiada precisión, puedes usar los convertidores incluidos en tu Commodore, que se han disfrazado habilmente como registros para paddles

Además te aconsejamos que leas la guia de referencia del programador de tu Commodore 64 ó VIC 20 si deseas más información sobre los circuitos integrados VIA y CIA.

COMMENTARIOS WORLD

Contabilidad 64

Versión B Precio, 29,500 Autor Jose Finto Espelt Propietarios y distribuidores: Casa de Software S. A. Aragón, 272, 8, 6 08007 Barcelona Tel, (93) 215 69 52

Este paquete de contabilidad se compone de un cartucho y un disco. La configuración ninna para su funcionamiento es un C-64 o SX 64, una unidad de disco (ya incorporada en el SX) y una impresora.

El primer punto a favor de este paquete es que puede utilizar muchos tipos de impresoras, desde las Commodore VC 1515 y MPS8/1, a las GP-16/6, las LPSON y otras, con lo que quien prefiera utilizar otro tipo de impresora, que se adapte mas a sus necesidades, puede hacerlo siempre y cuando la impresora pueda utilizar el protocoro Centronies

Litre las especificaciones del programa destacamos que puede controlar 600 euentas con 2.300 apuntes por diskette, y siempre se puede enlazar un disco con otro, pues los resultados anteriores son transferidos al nuevo disco que se atunza.

WORLD

ALICANTE

RARCELORIS

No soy un experto contable, pero con ayuda de amigos que lo son y conocen bien el tema, hemos utilizado el paquete y nos pareció muy bien y muy completo. Cualquier operacion contable está tenida en cuenta. El manda, bien presentado, conciso pero claro y perfectada na explicado. Mantenimiento de licheros, listado de fícheros y utilitarios.

Por otro lado, presenta la buena idea de dedicar un apartado completo a Nociones de Contabilidad para los "novicios", para quienes la contabilidad puede representar un cierto insterio. Igualmente la ultima sección del manual dedicada al Plan General Contable Español es una valiosa ayuda para expertos y novicios

El cartucho contiene el sistema de arranque del programa, son 8 K de EPROM que contienen unas rutinas en codigo maquina y el programa cargador en basic.

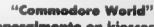
Cuando encendemos el ordenador pide que pongamos el disco, lo carga y arranea. Los programas estan escritos en basic, pero con rutinas en código maquina, de este modo aprovecha las ventajas de los dos (sencillez y velocidad a la vez). En realidad son varios programas, uno contiene el menu principal, y los otros desempeñan las diferentes funciones dasientos, diarios, consultas de cuentas, extractastentos, diarios, consultas de cuentas, extrac-

CONTABILIDAD
64

500 cuentos
2 000 apuntes
100 puntes
1

tos, sumas, saldos, situación y cierre, cuentas de explotación, balances, mantenimiento de licheros, listado de ficheros y un grupo de utilitarios)

Me parecio interesante la incorporación de los utilitarios que permiten copiar el disco, recuperar ficheros, etc. Expecialmente pensando en los datos que se pierden a veces en las (Pasa a pág. 66)



aparte de venderse generalmente en kioscos, se encuentra asimismo a la venta en las siguientes distribuciones de Commodore y librerías.



Bilbomicro SA, c/ Aureliano Valle, 7.
 48010 Bilbao, Telefono: (94) 443 43 51

HURGOS

F 1.5.A. c/ Madrid, 4. Telefono: (947) 20 46 24.
 CADIZ

 Video Computer, Comandante Gomez Ortega, 59. Algeeiras, Tel., (956) 65-39-02

CAMARIAS

 Relax. Rambia de Pulido, 85. Santa Cruz de Lenerife. Telefono: (922) 28 37 05.

CIUDAD REAL

 Electrónica Turrillo, c/ Pedrera Baja, 7 Feléfono: (926) 22 38 67.

CORUÑA, LA

- Cetronic, S.L. c/ Palomar, 22 Bajo. feléfono: (981) 27-26-54
- Photo Copy. c/ Teresa Herrera, 9 Telefono: (981) 21 34 21.
- Santusa, S. L. San Luis, 46 al 50. Teléfono: (981) 23 07 49
- Gesty Computer, Avda. Romero Donallo, 25
 Iel.: (981) 59 87 54. Santiago de Compostela.

GERONA

Digit Informatica, c/ Avda, 11 Setembre, 7.
 Olot, Telefono: (972) 26 94 01.

Vicrochip. c/Aigua, 3. Olot. Tel.: 26 36 63 • Regiscompte S.A. c/ Emilio Grahit, 17 Bts. Telefono: (972) 21 99 88.

HULLVA

 Computerlog, S.A. c/ Tendaleras, 15 Teléfono: (955) 25 81 99.

DIVER

 Kelson, Plaza España, S/N, San Antonio Abad, Telefono; (971) 34-13-09.

E E-OAL

Logdata. c/Burgo Nuevo, 4
 Iclefono: (987) 20 42 89

MicroBierzo, c/ Carlos 1, 2.
 Ieléfono: (987) 41 74 21. (Ponferrada)

MADRID

- . Chips. Clips. c/ Puerto Rico, 21-23.
- Flectrónica Lugo, c/ Barquillo, 40.
- Libreria García Peña. c/ Cavanilles, 52.
- Micromundo, SA, El Zoco, Vlajadahonda, feléfono: (91) 638 13 89.
- Pais de los Microordenadores, Bravo Murillo, 18, Fels.: (91) 446-33-17 - 446-37-12.

MALAGA

 Informatica Martinez SA, c/ Cristo de la Epidemia, 90. Felefono: (952) 26 15 60.

MURCIA

 Procoinsa, Ronda Norte, 27. 1eléfono: (968) 23 94 49.

PAMPLONA

 Microordenadores Ramar, c/ Navarro Villoslada, 7, Teléfono: (948) 23 72 80

SALAMANCA

Compusoft, c/ Arco, 1. Fel., (923) 21 59 93.

HAN BUILDSTIAM

 Donmiero, SA. c/ Arrasate, 6. feléfono: (943) 42 35 10

SANTANDER

 Libreria Hernández San Francisco, 15 Telefono: (942) 22-53-30.

SEVILLA

 Papeleria Mora, c/ Santa Cruz, 5, Ecua Felefono: (954) 83 14 80.

TARRAGONA

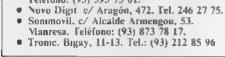
- Oficomplet. Plaza de la Cinta, 6 Tortosa. Teléfono: (977) 44 14 50.
- Comercial Informática de Tarragona SA c/ Gasómetro, 20. Teléfono: (977) 23 08 53.

VALLAGOUD

- Chips + Tips, S.A. c/ Juan de Juni, 3. Teléfono: (983) 33 40 00.
- kent Electronic, c/ Duque de la Victoria, 13.
 Tienda, 15. Telefono (983) 47 58 07

EARAGOZA

ADA Computer, Centro Independencia
 P^u Independencia, 24-26 Iel.; (976) 29 85 62



Casa Wagner, c/ Juan Carlos I, 37, Elda.

Ferreteria Progreso, c/ General Jordana, 28.

 Control Sistemas, Avda, Santa Vlarina, 25A. Lelefono, (924) 25-88-00.

Computer Service, Avda, Abad Marcet, 325.

Libreria Emilia Pérez Radua, c/ Mayor, 35.

Castellar del Vallés, Teléfono: (93) 714 89 51

Libreria Michel. Ronda Guinardo, 1. Sardañola.

Farrasa. Teléfono: (93) 788 63 77.

• Computerhard c/s. Jaime, 48. Granollers.

Gadesa, Les Valls, 12-14, Sabadell.

Mozart. c/ Jaime I, 145. Mollet

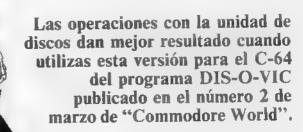
l'eléfono: (965) 39 03 96.

leléfono: (93) 870 09 19.

Telefono (93) 725 25 43.

Teléfono: (93) 593 75 01.

Elda. Teléfono: (965) 38 11 45.



Traducido por Valerie SHANKS y Adaptado por Diego ROMERO

Disk-0-64

Añade 13 Comandos Relacionados con Disco a tu C-64

En el número de marzo de "Commodore World" publicamos un programa de utilidades para disco para el VIC-20 llamado DISK-O-VIC. Resulta ser un programa muy útil que facilita mucho el manejo de la unidad de discos con el VIC-20. Yo queria conseguir lo mismo para el C-64, y por lo tanto decidí convertir el programa DISK-O-VIC en DISK-O-64

El principal problema era convertir las direcciones de las llamadas al sistema del DISK-O-VIC en las direcciones adecuadas para el DISK-O-64. Las llamadas Kernal resultaban sencillas, dado que son las mismas para los dos equipos y se han publicado en varios manuales de referencia. El verdadero problema eran otras llamadas al sistema como Warmstart, Reset y Printstring

Busqué en las ROM del C-64 para encontrar las rutinas que necesitaba. La l'abla I presenta los nombres de variables del listado original del DISK-O-VIC que necesitan cambiarse, junto con las nuevas direcciones del sistema para el C-64. Una vez conseguidas las direcciones correctas, la conversión en sí resulto bastante sencilla

Primero, encontré las llamadas que se tenían que modificar en los mismos sitios que se situaban en el listado original y localicé sus equivalentes en el volcado hexadecimal. Después, tuve que establecer los cambios necesarios para la pantalla de arranque. A continuación cargué el programa DISK-O-VIC, realicé los cambios necesarios con un monitor y salvé una copia de DISK-O-64 con la misma longitud y las mismas capacidades que DISK-O-VIC

Entrar el programa

Se necesita un monitor de lenguaje máquina para entrar el programa DISK-O-64 del listado del volcado hexadecimal Después de cargar y entrar el monitor, se empieza a entrar el programa en la dirección \$0801 y se continúa hasta \$0D2F. A continuación utiliza el método adecuado para salvar DISK-O-64 en tu monitor. Si utilizas el monitor del C-64 de Commodore, tienes que teclear

S"DISKO64",08,0801,0D2F

Asegúrate de que hayas utilizado \$0801 como la dirección inicial para que lo puedas cargar después como cualquier programa en Basic. Ahora dispones de una copia de DISK-O-64 que puede ser cargado y salvado como cualquier programa

en Basic. A continuación, sales del monitor, tecleas SYS64738 en el modo Directo, o bien apagas y enciendes el ordenador.

Ahora puedes cargar y ejecutar DISK-O-64 igual que cualquier otro programa en Basic Simplemente teclea LOAD "DISK-O-64",8 para cargar el programa en el ordenador y a continuación teclea RUN Si todo funciona como es debido, aparece la pantalla de arranque, y el programa DISK-O-64 se encuentra en su sitio, protegido en la parte superior de la memoria.

La Tabia 3 presenta los nuevos comandos. Si lees el artículo original sobre DISK-O-VIC, verás que todos los comandos son iguales para el DISK-O-64 Utiliza un disco para experimentar y prueba cada comando para poder familiarizarte con ellos (además de asegurar que todo funciona bien, y no existen errores de teclear)

Modificaciones de DLoad/DSave

Habia una cosa que me molestaba en el programa original DISK-O-VIC. No podía utilizar DLoad/DSave en un programa hibrido (es decir, un programa escrito en Basic y lenguaje máquina)

Parecia que se cargaba y se salvaba correctamente, pero me di cuenta de que el programa salvado tenía menos bloques

que el programa original.

El verdadero problema era que no me daba cuenta de lo que estaba pasando hasta que un día el programa hibrido DISK-O-VIC no queria ejecutarse, y lo tuve que volver a teclear todo. Con todo hay que decir que esta restricción fue mencionada en el artículo, y el fallo fue mío al no darme cuenta.

Descubrí que el problema venia de la rutina DLoad en el programa DISK-O-VIC. Esto ha sido modificado en el programa DISK-O-64 para que DLoad/D-Save puedan ser utilizados con programas hibridos con tal de que se carguen como un programa normal en Basic, es decir, LOAD"NAME",8.

La única excepción es el comando Append, que puede ser utilizado para añadir solamente programas escritos en Basic, y no programas hibridos. En realidad, la necesidad de añadir las partes escritas en Basic de programas hibridos no es muy frecuente (yo no lo he hecho nunca).

La Tabla 2 presenta los cambios para editar el programa original DIS-O-VIC para los que quieran modificar sus copias del programa.

Tellila I

Modificaciones de las rutinas para el DISK-O-64 sobre el listado del DISK-O-VIC

RUTINA	VIC	C-64
WARMST	C474	E386
WAIT .	C48C	A48C
INFIN	C49F	A49F
CHAIN	C533	A533
CLR	C659	A659
INTEGR	C96B	A96B
PSTRNG	CB1E	ABIE
ERROR	CF08	AF08
PRLINE	DDCD	BDCD
CHROUT	E742	E716
RESET	FD22	FCE2

Tabla 2

Modificaciones en el programa DISK-O-VIC para los comandos DLoad/DSave con programas híbridos.

DIRECCION	NEEVO
HEXADECIMAL	VALOR
DEL VIC	HEXADECIMAI
126E	86
126F	2D
1270	84
1271	2E
1272-127B	EA
127C	20
127D	33
127E	C5

Tabla 3

Explicación de los Comandos del programa DISK-O-64.

APPEND (Agregar).—Este comando permite que un programa en Basic que se encuentra en disco se agregue al final de un programa en memoria con números de linea más bajos. Su sintaxis correcta es

"APPEND "NOMBRE"

CATALOG (Catalogar).—Tecleas CA-IALOG para poder leer el directorio contenido en el disco sin destrozar el programa en la memoria COLLECT (Recoger).—Esto realiza

COLLECT (Recoger).—Esto realiza una operación de validación, es decir, que ordena el disco y hace que todos los bloques no utilizados sean disponibles.

DLoad.—Esto funciona igual que el comando normal de Load, pero no hace falta teclear ,8. Además, antes de cargar, inicializa el disco, y después lo verifica Se pueden cargar tanto los programas en Basic como los programas híbridos siempre y cuando se carguen los programas híbridos de la misma forma que los programas en Basic

DSave.—Igual que DLoad, sino que su función es la de salvar los programas en

disco

HEADER.—Este comando formatea el disco. Dado que todos los datos NIAD destruidos, te pregunta "ESTAS SI (AL RO? SI O NO". El sintaxis correcto est HEADER"NOMBRE NUEVO". Ixx. Tienes que emplear , l. Las letras xx representan cualquier tipo de identificación que le quieras poner (uno para cada disco)

INIT.—Esto es igual que OPEN15,8, 15,"I":CLOSE15 en Basic

KILL.—Esto es igual que apagar y volver a encender el ordenador o teclear

SYS64738.

OFF.—Este comando apaga el DISK-O-64, pero este y cualquier otro programa quedan en la memoria. El programa DISK-O-64 hace que el Basic vaya más despacio, así que es mejor quitarlo si te interesa la máxima velocidad. Para volver a encenderlo, teclea: SYS256*PEEK(56)+PEEK(55).

RENAME.—Te permite volver a nom-

brar un programa que ya existe en disco. Su sintaxis correcta es: RENAME"NOM-BRE ANTIGUO"TO"NOMBRE NUE-VO"

SCRATCH —Con este comando se puede borrar un programa del disc , su equivalente en Basic es OPEN15,8,15,150 NOMBRE".CLOSE15 También te pregunta "ESTAS SEGURO" \$1 0 NO". Su sintans conceta es SCRATCH"NOMBRE"

SIND (Enviar).—Este comando te permite enviar al disco cualquier comando que se pueda enviar en Basic, su equivalente en Basic es OPENIS, a., \$\frac{1}{2} \text{NXX}\text{CLOSE15}, donde xxxx representa mensaje que envias Su sintaxis (1) ecta es SEND "xxxx"

STATUS (Estado).—Con este comando se puede visualizar el estado del disco sin tener que ejecutar un programa. Cuando surge un error en disco, teclea STATI >

Listado del volcado hexadecimal

. 0800 00 40 88 01 00 99 22 93 . 0808 11 12 22 A3 31 35 29 22 . 0810 44 49 53 48 2D 4F 2D 36

. 0018 34 11 22 86 32 33 29 22 . 0820 43 36 34 20 44 49 53 4B . 0820 20 43 4F 4D 4D 41 4E 44

. 0630 20 50 41 43 48 41 47 45 . 0838 11 22 86 32 30 29 38 00

. 0840 78 06 02 80 99 22 42 59 . 0848 20 43 41 40 20 4F 56 45

. 0850 52 48 55 40 53 45 52 11 . 0850 22 86 31 38 29 22 46 52

. 0860 4F 4D 20 44 49 53 40 4F

(Pasa a pág. 53)

VERNOS A DUVICENTRO

SOLO ORDENADORES

(zcommodore

EN VALLADOLID KENT ELECTRONIC

> C/Duque de la Victoria, 13 Teléf. (983):39 65 84

SPECTROVIDEO

Accesorios, programas profesionales, juegos

APRENDE A PROGRAMAR POR ORDENADOR

Cursos completos y trimestrales

Clases muy reducidas, horario flexible

Profesores capacitados.

Cada alumno dispondrá de un ordenador

NUEVOS CONCEPTOS

C/ Magallanes, 1-1ºA - VALLADOLID Teléf. (983) 47 58 07

BASTORTH COBOL
LOGO FORTRAN

NUEVOS CONCEPTOS



MANUAL DEL MONITOR DE **CODIGO MAQUINA** PARA EL C-64 HES-MON (II)



G Ejecución

Empieza la ejecución de un programa en código máquina a partir de la dirección almacenada en el registro PC (contador de programa)

Ejemplos:

El HESMON va a la dirección que indica el Contador de Programa PC (ver-R) y empieza la ejecución del código máquina allí existente.

.G 3000

El HESMON pone la dirección \$3000 en el PC y empieza la ejecución del código máquina a partir de esta dirección.

H Busca en memoria

Básqueda de bytes que tengan cierto valor. El HESMON visualiza las direcciones que contienen un valor (o valores) determinado en una zona de memoria concreta

Etemplos:

H C000 D000 20 D2 FF

El HESMON chequea los bytes de memoria comprendidos entre \$C000 y \$D000 buscando la secuencia de bytes \$20, \$D2 y \$FF (JSR \$FFD2 en lenguaje ensamblador). Si encuentra tres bytes consecutivos que cumplan esta condición, visualiza la dirección del primero de ellos y continúa buscando, hasta llegar al limite indicado (en este caso \$D000).

.H C000 D000 "READ"

El HESMON chequea los bytes de memoria comprendidos entre \$C000 y \$D000 buscando la secuencia en caracteres ASCII correspondientes a "READ". visualizando las direcciones en donde la encuentre. La búsqueda puede interrumpirse pulsando la tecla STOP

I-Contenido ASCII

Visualiza el contenido en hexadecimal y en ASCII de una zona de memoria. Por ejemplo;

J F000 F020

Si no se especifica dirección final (segundo parámetro) el HESMON visualiza hasta el final de la memoria (SFI FF) o hasta que se pulse la tecla STOP

J=Salto a subrutina

Frecuta una llamada a una subrutina y vueive al programa principal con el modo W desactivado —durante la ejecución de la subrutina-sólo cuando está en modo W. Supongamos que estamos en modo W y en la pantalla se muestra el fragmento de código

147F LDA #\$00

1481 JSR \$A2C7

Se ejecuta la rutina que se inicia en \$A2C7 Cuando encuentra un RTS, volverá al modo W

L Load

Carga de zonas de memoria desde disco o cassette. Ejemplos:

L "RAM TEST" 01

Carga desde el cassette el programa llamado "RAM TEST"

.L "O:RAM TEST" 08

Carga desde el drive 0 del periférico 8 el programa llamado "RAM TEST". Los programas se cargan en la misma dirección de memoria en la que estaban cuando fueron grabados

M-Contenido hexadecimal

Visualiza el contenido de una zona de memoria en formato hexadecimal. Los bytes visualizados pueden ser modificados directamente situando el cursor sobre ellos, efectuando la modificación, y pulsando RETURN. Por ejemplo:

.M 1000 1005

.: 1000 43 0D 0A 43 42

.: 1005 0D 0A 7F 7F 7F

N=Transferencia de contenido

Traslada código máquina de una zona de memoria a otra. Se le puede indicar que las instrucciones de tres bytes, que apuntan a direcciones de memoria absolutas (JSR \$FCCO, CMP \$7540, X...) cambien su valor adaptándolo a la nueva posición que ocupen, o bien conserven inamovible el que ya tienen.

Ejemplos

N 7000 77FF 1000 0400 8000

Fraslada el código maquina comprendido entre \$7000 y \$77FF a partir de la dirección \$1000 (hasta \$17FF), Los dos últimos parámetros indican que las instrucciones de tres bytes que apunten a una dirección comprendida entre \$0400 y \$8000 deben cambiar su valor y adaptarse a la nueva posición del código máquina, por lo que en este caso se le restará \$6000 a cada una. Las instrucciones de tres bytes que apunten a otra dirección que no sea la comprendida entre \$0400 y \$8000 sc trasladarán manteniendo intacta su dirección.

Este comando se detiene cuando encuentra un código de ensamblado: erróneo (lo visualiza para indicar dónde se ha parado).

N 7000 77FF 0400 8000 W

El parámetro "W" al final indica que se está trasladando una tabla de bytes WORD, por lo que todas las direcciones (tomadas de dos en dos bytes) cuyo valor esté comprendido entre \$0400 y \$8000 se adaptan a la nueva posición de memoria

O=Desviación de salida

Este comando es el equivalente en MLM del comando BASIC CMD, Este permite al HESMON desviar la salida de pantalla hacia la impresora o hacia la unidad de disco

Por ejemplo:

.008 02 "O:TEST"

Donde 08 es el número de periférico (04 para la impresora) y 02 es la dirección secundaria (normalmente 02 a 0E para la unidad de disco).

Todos los parámetros son opcionales Si Vd. escribe sólo O automáticamente



HESMON abrirá el canal de impresora. Si vuelve a escribir O, volverá a las condiciones normales, cerrando el canal de comunicación con la impresora.

P Copia de pantalla

Este comando es una versión limitada del comando O. Copia el contenido de la pantalla en la impresora. Los parámetros de este comando son los mismos que para O, excepto que se desactiva automáticamente porque copia exactamente el contenido de la pantalla

NOTA: P no copiará gráficos de Alta

Resolución o Sprites.

Q Ejecución con puntos de rup-

Empieza la ejecución de un programa en código máquina a partir de la dirección que indica el PC (contador de programa). La ejecución es más lenta que la del comando, G, pero el Q puede detenerse mediante puntos de ruptura habilitados por el comando B.

Ejemplos:

O.

Inicia la ejecución del código máquina a partir de la dirección que indica el Contador de Programa (PC).

Q 1000

Asigna al PC el valor \$1000 y empieza la ejecución desde esta dirección.

R Contenido de registros

Visualiza el contenido de los registros según la siguiente clave:

PC: Contador de programa.

SR: Registro de estado.

AC: Acumulador.

XR. Registro indice X.

YR' Registro indice Y.

SP. Puntero del stack

Los valores de los mismos pueden ser cambiados situando el cursor sobre ellos y pulsando RETURN.

Ejemplo.

.R

PC SR AC XR YR SP 0000 01 02 03 04 05

Puede usarse este comando para asignar los valores del contador de programa, acumulador e índices antes de ejecutar programas mediante G, Q o W

RB Quitar puntos de ruptura

Eliminar los puntos de ruptura establecidos con el comando B. Solamente hay que especificar el punto de ruptura y será climinado.

Por ejemplo.

.RB 1050

El punto de ruptura existente en la posición \$1050 ha sido eliminado.

S Save

Grabación en disco o cassette de zonas de memoria. Ejemplos:

"S "PROGRAMA" 01 0800 0C81
Graba en el cassette, bajo el nombre de "PROGRAMA", el contenido de los bytes comprendidos entre las direcciones

\$0800 y \$0C80 Obsérvese que la última dirección a escribir en el penférico es la última más uno en el comando

.S "O:PROGRAMA" 08 0800 0C81

Graba en el drive 0 de la unidad de discos cuyo número de penférico es 8, bajo el nombre de "PROGRAMA", el contenido de los bytes comprendidos entre las direcciones \$800 y \$0C80.

T-Transferencia de memoria

Traslada el contenido de una zona de memoria de un lugar a otro de la RAM Por ejemplo

.T 1000 1100 5000

Toma los bytes comprendidos entre \$1000 y \$1100 y los sitúa a partir del \$5000.

Este comando no cambia ninguna dirección aunque se traslade código máquina, a diferencia de N.

U-Test de la RAM de color

U no necesita parámetros. Prueba la RAM de color e imprime OK cuando ha ejecutado el test. Si detecta un byte defectuoso, imprimirá su dirección.

V=Test de RAM

Este comando prueba un segmento de RAM. HESMON irá comprobando cíclicamente la sección de RAM especificada hasta que se pulse la tecla STOP (puede ser necesario pulsarla más de una vez). Cada vez que HESMON completa el test de la sección de RAM en pruebas, imprimirá un "." para que pueda ver que el comando está en acción

Si detecta un fallo en una posición de memoria, imprimirá la posición de memoria seguida de un número binario que muestra el dato que no ha sido guardado correctamente. Los bits de este número se muestran de izquierda a derecha (del bit más significativo al bit menos significativo). Los bits que presentan problemas se muestran con inversión de color.

Por ejemplo: .V 0400 0800

Verificará la RAM entre las direcciones \$0400 y \$0800 hasta que pulse la tecla STOP o ocurra un fallo.

W Ejecución paso a paso

Ejecuta una instrucción de código máquina a partir de la dirección indicada por el PC, mostrando el contenido desensamblado de la próxima instrucción a ejecutar. Al mismo tiempo también muestra el valor de los registros (R)

Ejemplos:

.W

Ejecuta la instrucción apuntada por el PC, muestra los valores de los registros, visualiza el desensamblado de la siguiente instrucción, y aguarda a que se pulse una tecla para ejecutarla. En el caso de que se pulse una tecla que normalmente tiene repetición (la barra espaciadora, por ejemplo) la ejecución se realizará al ritmo de esta repetición.

W.1000

Pone la dirección \$1000 en el PC y ejecuta la instrucción, muestra los valores de los registros, visualiza el desensamblado de la siguiente instrucción, y aguarda antes de ejecutaria,

X Salida al Basic

Sale del HESMON y pasa el control de C-64 al intérprete BASIC. Este comando tiene dos formas de actuación: La primera "XC" tiene el mismo efecto que desconectar el equipo excepto que el HESMON continúa conectado y puede transferírsele el control pulsando la tecla RESTORE. La segunda forma -"X"se comporta de forma similar a la pulsación simultánea de las teclas RUN/STOP y RESTORE en BASIC, no obstante su primera salida del HESMON debe hacerla con "XC", de lo contrario el BASIC funcionará mal. Una vez en BASIC para conseguir el mismo efecto que con RUN/ STOP y RESTORE sin HESMON, pulsar RESTÖRE, pulsar "X" y RETURN.

Conversión decimal a

Este comando imprime el equivalente hexadecimal de un número decimal. Si el número decimal es negativo mostrará el complemento a dos de un número de 16 bits hexadecimal y el número decimal positivo correspondiente.

Por ejemplo: # 1234

HESMON contestará en la misma linea; .#1234= \$04D2 1234

\$=Conversión hexadocimal a decimal

Este comando imprime el equivalente decimal de un número hexadecimal.

Por ejemplo: .\$ ABCD HESMON mostrará (en la misma línea): .\$ ABCD 43981

+ = Suma hexadecimal

Este comando imprime la suma de dos números hexadecimales en hexadecimal y decimal. Se deben usar cuatro dígitos hexadecimales

Por ejemplo: .+ 1234 5678

HESMON muestra.

1234 5678 = \$68AC 26796

-- Resta hexadecinal

Este comando imprime la diferencia entre los dos numeros hexadecimales en hexadecimal y decimal.

Por ejempio. .—1234 5678 HESMON mostrará

-1234 5678 = SBBBC 48060

Como aclaración final a la lista de comandos del HESMON debe tenerse en cuenta que los comandos que producen listas en pantalla (D, I y M) permiten que mediante las teclas de movimiento de cursor vertical se pueda prolongar el efecto del comando en las direcciones anteriores (con la tecla SHIFT) o posteriores a la última con la que se ha trabajado.



M STOLIFNWERK - Frankenberger Str. 32, 5100 AACHEN R.F. de Alemama

EL VIC-20 POR

DEN

TR

Algo de soft...
Una de las rutinas
mas importantes y
más sencillas del ROM
del VIC-20 es la rutina

comprueba tecla STOP pointer del programa JSR \$C82C 2C C8 7A 7B C7AE C7B1 LDA S7A A5 LDY \$7B A4 7B C0 02 ¿Modo directo? CPY #\$02 EA NOP BEQ \$C7BE FG 04 no: salvar pointer para CONT \$30 30 85 3E STY \$3F LDY #\$00 00 7A 80 coger carácter actual de pro. (\$7A) .1700 Pi1 ¿final de linea? 43 BNE \$C807 Dΰ Ãΰ 02 LDA (\$7A),Y В1 7A poner flag para END 18 BNE \$C7CE afinal pro 0.3 si: ejecutar END JMP \$C848 4C 4B C8 C8 LDA (\$7A),Y 7.8 STA \$39 85 39 memorizar no. de linea actual INY CR LDA(\$7A),Y 7A 81 STA SJA 85 3A TYA 9.8 pointer prg. a la siguiente lines 78 ADC \$7A D 65 85 7A STA \$7A BCC SC7E1 02 D INC \$7B E6 7B 6C 08 03 JMP SC7E4 JMP (\$0308) CHRGET coge siguiente carácte: 20 JSR \$0073 ejecutar instrucción JSR \$C7ED 20 ED C7 al principio de rutina JMP \$CTAD 4C AE C7 decodificar instrucción Bi final linea RTS (\$C828) BEQ \$C82B F0 3C C7FD #\$80 E9 80 SBC BCC \$C804 code del interpretador C7F1 CMP #523 C9 code de función o GO TO BCS SCROE BÖ 17 ASL DA AB В 99 OD CO LDA \$COOD, Y coger dirección de instrucció PHA LDA \$COOC,Y de la tabla B9 0C C0 PHA C800 48 coger siguiente carácter y eje-cutar instrucción a instrucción LET JMP \$0073 40 73 00 4C A5 C9 JMP \$C9A5 d comprobar siguiente instrucción CMP #\$3A C9 3A C807 BEQ \$C7E PQ D6 CR09 'SYNTAX ERROR' 4C 08 CF JMP \$CF08 6 comprueba code GO TO GO'-code menos 80 C9 4B CSOE CMP #S4B \$C80B CB10 DO P9 BNE CHRGET coge siguiente carácter 20 73 00 A9 A4-\$0073 JSR TO' code LDA #SA4 comprueba code 20 FF CE JSR SCEFF a la instrucción GOTO JMP \$C8A0 4C A0 C8

del interpretador de BASIC. Aqui se va a comentar de tal manera que no habrá apenas dificultades para implementar nuevas instrucciones al BASIC

La rutina CHRGET para coger el siguiente carácter del programa se ha puesto en la primera página de la memoria (\$0000-\$0100) porque así se ejecuta más rápidamente. Al encender el aparato se copia la rutina del ROM al RAM.

Algo de hard...

El circuito integrado más interesante del VIC-20 es el Video Interface-Chip

Para obtener resultados del ordenador nace falta una unidad de salida. El VIC-20 usa normalmente un televisor de color l'odas las señales necesarias para su funcionamiento las genera el VIC, el Video-Interface-Chip 6561. La única excepción es la señal de alta frecuencia a la que se modula la información. Esta frecuencia la genera el modulador externo. Todas las otras señales vienen del VIC.

Además posee este circuito integrado (1C) de 40 patillas dos conversores analógicos/digitales, cuatro generadores de frecuencia independientes y programables y una entrada para un Light-Pen. Pero primero vamos a ver el significado de

		_
cada	. patiila.	
Pin	Desc.	Funcion
1	N C.	sin función
2	COMPCOLOR	salida, información sobre color para TV
3	SYNC&LUM	salida. Información sobre sincronización y bri lo para 1V
4	VR/W	entrada, lectura/escritura
- 5	VDH	bit de datos II
hast.	a	
B	VD8	bit de Jatos 8
9	BD7	bit de datos 7
hast	a	
16	BD0	bit de datos 0
1.7	POT X	entrada, conversor A/D
18	POT Y	entrada, conversor A/D 2
19	COMP SND	salida sonido
	Vss	alimentación masa
21	A0	bit de dirección 0
hast	d	
34	A13	bit de dirección 13
35	P01	saiida ciclo
36	P02	salida ciclo, sin usar
37	LITE PEN	entrada lapiz óptico
	02un	entrada frecuencia de

01in

Vdd

39

40

entrada frecuencia de

ammentación +5V

chargo



Al observar las patillas del VIC resalta el bus VD8 hasta VDII, con lo que dispone de un bus de datos de 12 bits Esto resulta algo extraño. Pero la respuesta es tan fácil como convincente. Los cuatro bits adicionales sirven para definir el color del caracter. Y así también se explica la posibilidad de 16 colores, porque con 4 bits se pueden hacer 16 combinaciones (...)

Las restantes 8 líneas de datos van al bus de datos BD. Este bus tiene una doble función. Cuando el ciclo del sistema estáen H lo usa el procesador para acceder al RAM. En el estado L lo usa el VIC. Ahora el VIC lee el Video -y color-RAM, y salva los datos necesarios para la representación de los caracteres del Character ROM

l'ambién las líneas de direcciones del VIC tienen una doble funcion. En el tiempo que el procesador ocupa el bus de direcciones tambien direcciona a los 16 registros internos del VIC.

Las entradas POT X y POT Y están conectadas a las patillas 5 y 9 del Controller-Port para el scanning de los

Paddles. Su resistencia tiene que variar entre 200 y 500.000 ohmios. La entrada del lapiz óptico va directamente a la patilla 6 del mismo port.

La señal de baja frecuencia de los cuatro generadores (patilla 19) se filtra para poder acceder a ella por el contacto 3 del Video/Audio-Port. Se tiene la posibilidad de mezclar otra señal de audio, p. ej., un micrófono, que sólo tiene que conectarlo al contacto Y del Expansion-Port y fijarse de que la señal tenga unos 3 Vss



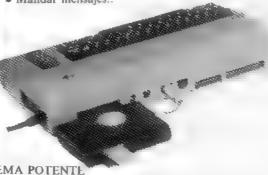
¿Que no sabes cómo suscribirte a Commodore World? ¡¡¡Pues venga, llámanos!!! (91) 231 23 88/95 y (93) 212 73 45



RED LOCAL PARA COMMODORE 54

POSIBILIDADES:

- Más de 20 puestos de trabajo.
- Utilización de todos los periféricos de la gama profesional Commodore.
- Conexión de todos los puestos sobre mismas unidades periféricas a través BUS IÈEE.
- Diálogo bidirecciones entre puestos.
- Manejo a distancia de cualquier ordenador.
- Posibilidad de coger y pasar una pantalla.
- Cargar un programa a todos los puestos a la vez.
- Mandar mensales...



- SISTEMA POTENTE
- FACIL DE MANEJAR, NO PRECISA SOFTWARE
- ADAPTABLE A LAS CONDICIONES MAS EXIGENTES
- APLICACIONES EDUCATIVAS, COMERCIALES...

COMPARTIENDO EXPERIENCIAS ENTRE AMIGOS



E sta sección está dedicada a la colaboración de todos nuestros lectores y está dividida en dos partes:

1) Programación: Programas y similares.

2) Magia: Trucos, sugerencias, etcétera (pág. 48)

Habrá premios y alicientes para todos los participantes (ver pág. 5).

Enviarnos vuestra dirección para que podáis poneros en contacto unos con otros. ¡Anímaros, chicos (...y chicas)!

Sabotaje Nuclear

C-64

Francisco Sarrias Pedemonte Laforja, 100, 2º 3º e-B Barcelona-21 El grupo terrorista MOPILETTE ha saboteado los silos de almacenamiento de plutonio concentrado Estos silos si estallan

provocan una reacción en cadena que destruiría la humanidad Su misión como jefe de seguridad de la central consiste en penetrar en las distintas habitaciones del complejo y cerrar las válvulas numeradas según el orden que indiquen las claves de

desactivación
Pero atención: si desactiva las válvulas en un orden que no es
el indicado, usted es destruido y las esperanzas de la humanidad

Una advertencia final, las habitaciones contienen gases letales y usted na de consumir el oxígeno que hay en su vehículo que va disminuyendo con rapidez y si llega a extinguirse usted perece

Descripción del Juego:

Al comenzar el juego se encuentra usted en su vehículo esférico en una enorme sala donde están marcadas las válvulas de seguridad. En la parte inferior de la pantalla está indicado el orden en que usted debe pasar por los números para desactivar las cargas situadas en esa habitación. Para mover su vehículo utilice las siguientes teclas.

Y - arriba N - abajo G - .zquierda J - derecha

Al comenzar la partida, usted debe seleccionar su nivel de Juego que determinara la rapidez con que su reserva de oxígeno se consumirá, así como la puntuación de BONUS que usted reciba al final.zar la pantalla

Al cerrar una válvula el ordenador comprueba si ésta seguia la cadencia indicada en la clave de la habitación, de ser así ese número se coloca en rojo en la clave y usted procede a pasar por el siguiente número, de lo contrario usted es destruido y pierde una vida de las tres que se le conceden al iniciarse el juego

Otra forma de perder la vida es cuando se le agota la reserva de oxigeno —que inicialmente es de 999 litros—; cuando el oxigeno es muy escaso su marcador de oxigeno se volverá loco Tanto el nivel como las vidas que se tienen se incrementan automaticamente a medida que se va adquiriendo una cierta puntuación, así pues el nivel se aumenta más o menos cada 5.000 puntos —digo más o menos porque igual aumenta 2 ó 3 unidades de golpe, forma parte del riesgo y la aventura. Y las vidas aumentan cada 15.000 puntos

Según el nivel en que se halle el oxígeno disminuira más o menos lentamente y usted tendrá que dirigir su vehículo con mucha habilidad para aprovecharlo al maximo —piense que su vehículo esférico es un modelo anticuado y de velocidad limitada

Al conseguir cerrar todas las válvulas de una sala, su puntuación se verá incrementada —BONUS— en función del nivel y del oxígeno que aún le resta

Sólo me queda advertir que yo personalmente aun no he pasado del nivel seis

Descripción del Programa:

Si bien el programa no goza inicialmente de una espectacularidad desbordante, resulta bastante adictivo y convida al jugador a probar de superar su anterior puntuación

El programa carece de efectos de sonido y de otros utiliajes de costumbre en la "full arcade action" tales como la utilización de rutinas en "machine code" para dar rapidez a los movimientos. Pero mi objetivo era dar una idea básica que contuviera los menos "pokes" posibles y que fuera rápida y sencilla de copiar. Sería interesante observar los adornos que cada cual añada al juego y comparar los resultados finales

Algunas implementaciones sencillas son la aparición de minas, habitaciones creadas por random, utilización de "esprites" en los gráficos, sonidos, tablas de records, paredes de la habitación electrificadas, monstruos que te persigan, etc.. cada cual con su imaginación y habilidad para la programación.

Refiriéndonos al programa en si, podemos observar en listado que está organizado en una serie de subrutinas de utilidad que son llamadas desde el programa principal.

100-130 Subrutina de presentación y final.

200-398 Subrutina que dibuja en pantalla el trazado de la habitación, los marcadores y la clave de desactivación 400-420 Subrutina que busca una posición al azar y la coloca

en la variable X.

32/Commodore World Octubre 1984

450-495 Subrutina que se encarga de generar la clave de desactivación de cada habitación.

Subrutina que se encarga de gestionar la muerte del 500-535 individuo, comprueba si quedan vidas y en caso negativo finaliza la partida y propone la revancha

600-615 Subrutina que se encarga de añadir los BONUS necesarios al marcador cuando se ha completado una pantalla,

700-720 Subrutina que se encarga de elevar el nivel de juego o las vidas si procede

1000 Borra la pantalla y coloca los colores de fondo. 1010-1030 Inicializa la partida y pregunta el nivel inicial de

1060-1070 Inicializa variables y pinta la habitación.

1080-1110 Coloca al azar las válvulas a lo largo de la habitación

1120-1140 Se encargan de calcular la posición del vehículo según las teclas que se hayan pulsado.

1143 Comprueba si el vehículo ha cerrado una válvula.

2 WEM SHBOTAJE NOLLEHR

3 KEM

REM AUTOR : FRANCISCO

5 KEM SARRIAS PEDEMONTE 6 REM

7 REM PARH EL CBM=64

8 REM

10 GUTO 1000

100 FOR I=1 TO 100

110 PS=INT(RND(1)*975) - CD=INT(RND(1) #255

120 PUKE1824+PS.B1 POKE 55296+PS.CO 130 NEXT I RETURN

200 PRINT"[[LR]####NIVEL###VIDHS###P

UNTUSH来和UNIDENDER中央电台

21d PRINT"#####";NI;"#####";VI;"####

非",SC。"郑卓卓来中亚",OX。

215 PRINT

AND PRINT [WHI IT 400 MM+1 1

305 PRINT"[CCMM+][BSPC3[CUMM+3[145FC]

[CONM+3[14SPC][COMM+3*

310 PRINT"[CUMM+][BSPC][COMM+][148PC]

ECOMM+JE145PC_ECUMM+1

315 PRINT"[COMM+][85PC][CGMM+][145PC]

[COMM+][145PCJ[coMM+]

320 PRINT"[500MM+][4SPC][60MM+][295PC] (U0MM+0"

325 PRINT"[CUMM+][385PC][COMM+]"; 330 PRINT"COUMM+JL38SPCJECOMM+1"

BBS PRINT"[COMM+][BSPC][COMM+][29SPC] I CUMM+14

340 PRINT'[COMM+][85PC]ECOMM+][145PC]

LFUMM+10145P01LCOMM+1

345 PRINT"[COMM+3185PC]ECOMM+3[145PC]

LCOMM+31145PC31COMM+31 350 PRINT"[COMM+3055PC][10COMM+308SPC]

[COMM+][7SPC][/COMM+][4SPC][2COMM+]"

355 PRINT"[COMM+][235PC][COMM+][75PC] ClumM+)[63P]][CuMM+)', 360 PRINT"[CGMM+][22SPC][6COMM+][3SPC]

[60MM+][65P6][60MM+]

365 PRINT"LCOMM+1[178PC][58PC][COMM+]

[8SPC][(OMM+][6SPC][COMM+]".

370 PRINT"[COMM+][22SPL][COMM+][5SPC] [35P6][COMM+][65P6][CUMM+]"

375 PRINT"[COMM+][4SPC][6COMM+][12SPL] (comm+)[85P6][Comm+][65P6][Comm+]",
S80 PRINT"[Comm+][95P6][COMM+][125P6]

[COMM+108SP0][LOMM+1(65PU][COMM+]" 385 PRINT"[COMM+JE95PC][COMM+][BSPU] [SCOMM+][8SPC][COMM+][6SPC][COMM+]",

390 PRINT"[CUMM+][9SPC][COMM+][21SPC] [GOMM+][BSPC][CGMM+]] 395 PRINT"[40COMM+1",

1145-1200 Disminuye el oxígeno y actualiza los marcadores. 1210-1240 Comprueban si la válvula sigue la cadencia de la clave y genera la puntuación y la presentación en

pantalla correspondiente.

Variables principales

PS y CO son utilizadas para la presentación

NI almacena el nivel de juego

VI almacena las vidas

SC almacena la puntuación

OX almacena el oxigeno

PI, P2, P3 utilizadas para los "peeks"

X, I, K utilizadas en labores menores

CL\$ almacena la clave

CN es utilizado para determinar si la cadencia de desarticulacion es correcta

SD antigua posición vehiculo

OSD actual posición vehículo

CL utilizada en la elaboración de la clave.

398 PRINT"[RV5UN][YEL][45PE][LHVE[5PU]

"",CL#,"[10SPC]" RETURN

400 X=INT(RND(1)#800)+120

410 IF PEEK(1024+X)(2)32 THEN 400

420 RETURN

450 CL\$="".FOR I=1 TO 9

455 CL(1)=INT(RND(1)*10)

460 FUR K#0 TO I-1

465 IF CL(I)=CL(K) THEN 455

470 NEXT K

475 CL#(I)=STR#/CL(I)/

480 CL\$=CL\$+CL\$(I)

485 NEXT I

POR FIN!

EL SOFT DIDACTICO Y RECREATIVO DE MAS EXITO EN U.S.A. DISPONIBLE EN NUESTRO MERCADO

IdeaLogic, S. A es una nueva empresa dedicada a la distribución y manulatturaci il de software y accesorios para microordenadores que ofrece, a partir de este mes de octubre, una innovadora gama de productos que sin duda dará mucho que habiar

Distintas marcas de reconocido prestigio internaciona, y de gran éxito en el mercado han firmado contratos en exclusiva con Ideal.ogie, para que esta las

represente en el mercado español

IdeaLogic contará con su propia red de distribución que cubrirá tambien las Islas Canarias y Andorra. Dicha red está en fase actual de negociación Los primeros títulos que aparecen vienen a cubrir un importante hueco

Los primeros títulos que aparecen vienen a cubrir un importante nuevo existente en los actuales catálogos de software. Nos referimos a la pareca del software educativo y didactico. El líder mundia, en este campo es, sin du l. Infirma de Boston SPINNAKER SOFTWARE CORP, cuyas ventas representant el 40% del mercado americano. De esta firma IdeaLogic anza cinco itulos MIL CARAS, TECLAS DIVERTIDAS, COMPULANDIA, EL RANCHO y un nuevo producto que ya ha sido denominado como el "MICHAEL JACKSON DEL SOFTWARE", este es ALE EN LAS GRU-TAS DE COLOR Todos estos programas han sido cuidadosamente creados por especialistas en la educación, en los gráficos, la musica y la programa ción, con un objetivo central el mão, que a la vez que juega con ej ordenador

aprende
Otra marca muy ligada e interesada en el mundo del niño es F-P Learning
Software, sus titulos son de muy reciente aparición en los Estados l nidos y
ya han recibido innumerables elogios. De su catalogo IdeaLogie presenta en
un primer momento LABERINTOS LOGICOS, BAILE MOS Y NUMEROS
LOCOS. Estos divertidos programas son realmente encantadores y hacen
realidad el que el microordenador puede ser usado como berramienta ludica a
la vez que el microordenador puede ser usado como berramienta ludica a

La vez que educativa y creativa

La software de IdeaLogic aparece para Appie IIe, para el cual está desarrollando un largo catalogo de software educativo, y también para los microordenadores Commodore 64 y Sinciari Spectrum Para los usuarios de. C 64
Ilega MUSICALC 1, 2 y 3, con el cual es posible convertir su Commodore en
una auténtica orquesta, componiendo canciones, modificando as. añadiendo instrumentos y experimentando creativamente con la musica. Todo ello aun sin poseer conocimientos musicales. Su única limitación es la imag nación del usaano

Otras importantes novedades son la Valiant Turrie, robot mecanico para ser usado desde el lenguaje LOGO, la tabia grafica de bajo precio SUPLR SKETCH y el irrompible joystick de precision SUPER STIK.

Por fin, y como novedad de desarrollo propio, presenta an mini ir en color de media-alta resolución para ser usado tanto con micros qui dan seña. RGB o los que ofrecen señal compuesta. Su precio es muy intercapite lidealogic tiene preparadas otras diversas e interesantes novedades que la margia programa propria al mercado. Sus oficiais están en foran y a de Carlis.

lanzará proximamente al mercado. Sus oficinas están en Gran V a de Carlos III, 97K, 08028 BARCFLONA Tel 330 33 08*

490 CN=0 REM CONTADOR NUMERO 495 RETURN 500 VI=VI-1 IFVI>000T0535 505 GUSUB100 515 PRINT"[CLR][80kSkD][60RSkR]" 520 INPUT 'BESENSISPOJOTRAISPOJPARTID ALSPOISZN**, PH\$ 525 IFPAS="N"THEN END 538 RUN 535 6010 1055 600 FORT=UKTO @ STEP-100/NI 605 SE=SE+NI#100 GOSUB200 608 60508700 618 NEXT I 615 80701965 700 IFSC/5000>INT(SC/5000)HNDSC/5000 CINTUSC/5000/#1.05THENNI=NI+1 710 IFSC/150000 INT(SC/15000) HNDSC/15 ODUCINTOSUZISUUU:#1.UZTHENVI=VI+1 728 RETURN 1000 PRINT"[CLR]" PUKE53280.0 PUKE5 3261.0 1810 GOS JB108 1020 PRINT"[CLR][[UCRSRU][45PC]". 1030 INPUT"[SPC] INTRODUCE[SPC]NIVEL[SPC] 1-5", NI PRINT"L. LK. " 1060 VI=3 SC=0 DIMCL*(9), CL(9)

1065 0X=999 60SUE450 60SUE200 1070 GUSUB 400 SD=X OSD=SD 1080 FOR I=49 TO 57 1090 GUSUB 400 REM X ALEHTORIA 1100 POKE1024+X, I - POKE55296+X, B 1110 NEXT I 1120 PI=PEEK(197) P2=1024+5D 1130 SD=5D+1*(P1=26 HND PEEK(P2-1)() 102)-1*(P1=34 HND PEEK (P2+1) (>102) 1140 SB=SD+40*(P1=25 HND PEEK(P2-40) O102)-40#(P1=39 RNB PEEK(P2+40)(>10 1143 P3=PEEK(1024+5D) IF P3)48 AND P 3K58THEN 1210 1145 OX=UX-NI IFOX<0THENGUSUR504 1150 POKE1024+05D.32 POKE1024+5D.61 บรม=รมิ 1190 PRINT"[HUM][LRSRD][55PC]",NI,"[58Pc] ".VI: "[5SPC]".SC: "[6SPC]".OX: 1200 GOTO 1120 1210 CN=CN+2 IFCN=18THENGOSUB600 1220 IFP3-48<>VHL(NID\$(CL\$,CN,1))THE N. GOSUBSMA 1230 SC=SC+INT(RNB(1)#25)#10 1235 GUSUEZWM 1240 POKE56186+CN, 2 UOTO 1145



Laberinto

VIC-20

Carlos Rojas López Gustavo Gallardo, 7. 4ºD Sevilla-13

que escribo a la revista. enviando un programa Sin embargo sería tan malo que ni lo han publicado. Espero que

Esta es la segunda vez

esta vez no ocurra lo mismo

Les mando un sencillo programa de mego; tal vez algunos opinen que es una versión del popular Pack-Man, pero yo opino que es muy original.

El juego consiste en mover al packman comiéndose los puntitos con el joystick; verás los puntos en la merte superior izquierda, pero debes tener cuidado con los monstruos que te acechan. En la parte superior derecha hay un cronómetro que te indica los segundos que faltan para que los monstruos apa-

rezean. Cada vez que salgan tardarán en aparecer un segundo menos, hasta los 15 segundos, en el que retornará a los 30 iniciales

De vez en cuando saldrá una cereza que vale 500 puntos. Cuando consigas 2.200 puntos, puedes comerte el signo gráfico de arriba a la izquierda.

Cada 7,500 puntos tendrás un packman más —tienes 3 al principio—. El juego termina cuando agotes tus tres packman

NOTA: los monstruos te comen cuando coinciden con tu packman o con las bombas que deja tras una aparición completa de todos los monstruos.

Como podéis comprobar, el juego es muy sencillo y está dividido en dos partes, o sea, grabado en dos partes:

1) Esta parte crea los gráficos de alta resolución que definen las figuras o muñecos que aparecerán durante el juego. Consta

GLOSARIO A.-Amperio. analógico, asíncrono, sumador (ad ABEND.—(ABnormal END of (ask), es la terminación de una función o programa antes de su tiempo habitual, suele ser debido a un error producido durante la ejecución que no ha podido ser detectado. Abort.-Es la terminación de una

rutina dando un mensaje de error detectado por el sistema.

Absolute error.—(Error absoluto), es el rsultado de restar el valor obtenido o medido del valor teórico esperado.

A bus .- (Address bus), es el bus de direcciones de nuestro ordenador

AC .- Corriente alterna (alternating current) o acoplador acústico (acoustic coupler)

ACC.—Acumulador.

Access.-(Acceso), es la manera en que nos referimos a los archivos o bloques de datos (directo, inmediato, serie, etc.).

Access time.—(Tiempo de acceso), es el intervalo de tiempo desde que la CPU solicita un dato hasta que lo lee (depende de la velocidad de la memoria o periféricos)

ACIA.—(Asynchronous communications interface adapter), es el dispositivo o chip que nos permite comunicarnos con un bus de comunicaciones serie asincrono (por ejemplo con un modem o impresora rs-232).

ACK .- (Acknowledgment), es la señal de reconocimiento, sirve para que un dispositivo indique al otro que ha recibido los datos que le envía.

Acoplador acústico.-(AC), es el dispositivo que permite convertir las señales enviadas por un modem a todos audibles para que estos sean transmitidos por el micrófono del teléfono a la línea telefónica y viceversa

REPRODUCCION DE LISTADOS IMPRESOS DEFECTUOSAMENTE EN EL COMMODORE WORLD DE SEPTIEMBRE

Es la "Hora" de aprender, pág. 15 C-64

Ø REM PARA EL RELOJ-064

```
1 L#= 132714091529168917671763176016
781516139813201323
2 LH##"1329144+152917691762176317581
63815181313137013231
3 IH$="+3978+3978-0145-4177-4177-406
6-3978-3978+0145+4177+4177+4066
4 IP$='+3978+3978-0164-4177-4177-406
6-3978-3978+0167+4177+4177+4066
5 B$="124312491331153117711849184318
381756151613161238" S2=54285
6 CO=54272 V=54296 SC=53281 BO=53281
 M2=54283 - R2=54284 H2=54288 * L2=54279
8 FORX=54272T054296 POKEX,0 NEXTX PO
KEY, 15 POKER2, 116 POKES2, 128
10 POKESC, 12: POKEBO, 3
   PRINT"[CLR][5CRSRR][COMM4][RVSON]
[SPC]DIAGRAMA[SPC]DE[SPC]HURA[SPC][RYSOFF]
  PRINT"ECRSRD3", SPC(8), "00" . PRINT"E6SPC3
[500MM@][SPC15"
15 PRINT"[35PC]55E5HIFTN][25PC]12E8PC]
(SHIFTM)" PRINT"(45PC)(SHIFTN)11(35PL)
105PC103H1FTM103PC119" PRINT"05PC150
[SHIFTH][95PC][SHIFTM]"
20 PRINT"[2SPC][COMMM]10[SPC]RELOJ[SPC]
2[SPC][COMMG]" PRINT"[2SPC][COMMM]".
SPC<117, "ECONMOD"
25 PRINT"45[COMMM]9[2SPC][SHIFTQ][SHIFTW]
[COMM+184SPC13ECOMMG]15" PRINT"[25PC1
[COMMM1[5SPC][SHIFT-][5SPC][COMM8]"
30 PRINT"[25PC][COMMM]8[4SPC][SHIFT-]
[3SPC]4[SPC][COMMG]".PRINT"[SPC]40[SHIFTM]
[45PU][SHIFT-][45PC][SHIFTN]20
35 PRINT 'L45PC1[SHIFTN]ESPC17[SPC1[SHIFTQ]
[SP035[SP03CSHIFTN3" PRINT"[35P0335[SHIFTM3
[2SPC]6[2SPL][SHIFTN]25"
40 PRINT"[6SPC][5COMMT]" PRINTTAB(7)
"30"
45 PRINT"[CRSRD][48PC]HORA=[SPC]9 30
50 PRINT'IZCRSRDJESPCJPULSAISPCJ'F7'
(SPC)PARH(SPE)COMENZHR(SPC) J(ZSPC)P
ULLH" PRINT"(SPC]/1/[SPC]PARA[SPC]IN
STRUCCIONES."
55 GETS# IFS##""THEN55
56 IF5$="I"THEN58
57 GOTO280
58 PRINT"[CLR]", SPC(11), "[RYSON][SPC]
INSTRUCTIONES(SPCTERVSOFF)"
78 PRINT'(CRSRD)(SPCTGANAS(SPCTACERT
ANDDESPECISESPECIALES."
75 PRINT"[2CRSRD][RYSON][SPC]TECLAS[SPC]
DECSPOJCONTROLESPOJERVSOFFJ":PRINT"[2CRSRD]
'F1'ESPCIMUEVEESPCILAESPCIMANILLAESPCI
CHICA.
80 PRINT"[CRSRD1'F3'[SPC]MUEVE(SPC]L
ACSPOIMANILLACSPOIGRANDE."
85 PRINT"[CRSRD]'F7'[SPC]PARACSPC]EL
ISPOIRELOJ.'
```

87 PRINT"(30ASRD)(SPU](PULSA(SPC)UNA

```
(SPCITECLA.)"
90 GETS$ IFS$=""THEN90
 150 POKESC, 11 POKEBO, 7 PRINT"[CLR][WHT]
 CORSEDICAVSONIDIFICULTADESPOIDECSPOI
LAISPOJPRUEBAERVSOFF]"
 155 PRINT"[CRSRD](1)[SPC]HURAS" PRIN
 T"[CRSRD](2)[SPC]HDRASISPC]Y(SPC)1/2
-HÜRHS"
160 PRINT"[CRSRD](3)[SPC]INTERVALOS-
SISPCIMIN. "
165 PRINT"(6CRSRDJ(25PC)PULSA[SPC]1;
2.0[2SPC]3" PRINT"[CRSRD][SPC]0[2CRSRD]
[2SPC]PULSAESPC]'F'[SPC]PARAESPC]FIN
170 GEIDS IFDS##UTHEN170
173 IFD#="F' THENPOKEY, 0 POKESC, 6 POK
EBO, 14 PRINT"[CLR][COMM7]" : END
175 DMVAL(D#)
180 IFD>30RD<1THEN170
200 PRINT"(CLR](RED)" R=0.W=0 POKESC
, B. POKEBO, 2
210 PRINTTAB(12)"[1500MM@]" PRINTTAB
(10)"[RVSON][SHIFTE][RVSOFF][SHIFTE]
";SPC(15);"[SHIFTM]"
211 PRINTTAB(9)"[RVSON][SHIFTE][RVSOFF]
[SHIFTE](SPC]11(SSPC)12(65PC)11(SHIFTM]
  PRINTTAB(8)"[RVSON][SHIFTE][RVSOFF]
[SHIFT£]", SPC(19), "(SHIFTM]"
212 PRINTTAB(8) "[RYSON][SPC][kVSOFF]
10". SPC(18,, '2[SHIF IM]" PWINTTHB(8)"
[RVSCN][SPC][RVSOFF]".SPC(22),"[COMMG]
213 PRINTTAB(B)"[RVSON][SPC][RVSOFF]
 ",SPC(22),"[COMMG1" PRINTTAB(8)"[RVS.IN]
[SPC][RYSUFF]", SPC(22), "[UCM/46]"
214 PRINTTAB(8)"[RVSON][SPC][RVSOFF]
">SPC(22);"&COMMG1" PRINTTHB(8)'(RV5CN)
[SPC][RVSGFF]">SPC(22)>11 DMMG) 1
215 PRINTTABOR "CRYSON I CSPC I KYSOFF)
 ,SPC(9), "[RVSON][4SPC][RVSOFF]", SPC
(9), "ECOMM31"
216 PRINTTAB(8)"[RVSON][SPC][RVSUFF]
[SPC19", SPC(7), "[RV UN][4SPC][RVSOFF]
 SPC(7), "BESPCICCEMMG1"
217 PRINTTAB(8) "[RV50N][SPC][RVS0FF]
 ,SPC(9), 'CRYSONIC4SPCIER/SOFFI", SPC
(9), "[COMMG]"
218 PRINTTAB(8) "[RVSON][SPC][RVSOFF]
',SPC(9),"[RVSON][45PC][RV,DFF]",SPC
(9), "CCOMM61"
219 PRINTTAB(8) "[RYSON][SPC][RYSOFF]
 SPU(22), "ICOMMGT" PRINTTAB(8,"[RVSUN]
LSPCILRySOFFI", SPC(22), "[), UMMG]"
220 PRINTTAB(8) "[RVSON][SPC][RVSOFF]
 SPU(22), "(COMMG)" PRINTTAB(8)"[RVSoN]
LSPCJLRY50FF]",SPC(22),"[CUMMG]'
221 PRINTTAB(8)"[RVSUNJ[SPC][RVSOFF]
",SPC(22),"[COMMG]" PRINTTAB(8)"[COMM*]
ERVSONICCOMM#JERVSOFF18'\SPC(18)\"463HIFTN1
222 PRINTTAB(9)"[COMM*][RVSON][COMM*]
LRVSOFF1", SPC(18), "[SHIFTN]" PRINTTH
B.10/'[COMM#][RVSGN][COMM#][RVSGFF]7
' SPC 6)/'6",SPC.7/,'5[SH]FTN]"
```

223 PRINTTAB(11) 'CCCMM*1ERVSON1ECOMM#]

[RVSOFF][14COMM@][SHIFTN]"

240 HR=INT(RND(1)#12+1)-0=0 242 IFD=3THENMN=INT(RND(1)#12).GOTO2 48 243 IFB=2THENMN=6#INT(RND(1)#2):GOTO 248 247 MN=8 248 IFMN#BURMN#1THENPRINT"[HOM3",HR, " @",5*MN GOTO250 243 PRINT"[HUM1", HR, ", ", 5*MN 250 He1 : RM=0 251 IFRMD2THEMPL=VHL(MID#(LH#,(H-1)# 4+1,4)) 6070280 261 IFRMD11THENH=H+1 RM=@ IFH012THEN250 262 270 PL=VAL(MID#(L#,(H-1)#4+1,4)) 260 PB=VAL(MID\$(B\$,4*FM+1,4)) 281 PPINT"[HOM][14CRSRR][12SPC][10CRSRL] " H S#RM, 284 IFRM=@THEN11=4@ PP=66 GUTO29@ 285 II=VAL(MID#(IP#,(RM-1)#5+1,3)) 286 PP=VHL(MID*(IP*,(RM-1)*5+4,2)) 289 [F(RM)11)OR(H)12)THEN261 290 POKEPB. 160 PUKEPB+CO. 0 300 FORK=1105 POKEPB+K#II, PP POKEPB+ K#II+UU-U NEXTK SUS POKEH2.30 PONEL2,141 POKEW2,17-F ORK=17010 NEXTK POKEW2.0 310 IFRM>2THENI=VAL(MID\$(IH\$,(H-1)45 +1,3)).P=VAL(MID*(IH*,(H-1)*5+4,2)): GOTU330 320 I=VAL(MIL*(IP\$,(H-1)*5+1,3)):P=V ALCMID#CIP#JCH-1 #5+4 2. 330 POKEPL, 160 PUKEPL+CO, 6 340 FORK=1TO3 PCKEPL+K#I;P POKEPL+K# I+CO/6 NEXTK 358 FORK=1T0688 NEXTK 353 IFQ2=1THEN650 355 GETS# IFS#="1THEN379 357 [F3#='[F1]"THENWM##M+12 SUT0379 358 IFS\$="[F3]"THENRM=RM+2 G0T0379 359 IFHR=HANDMN=RMTHEN500 360 GOTD680 379 POKEH2,61 POKEL2,126 POKEW2,17 F ORK=IT010 NEXTK FUKEW2.0 380 POKEPB, 32 FORK=1TO5 PUKEPB+K#II, 32 · NEXTH 381 POKEPL,32 FORK=1TO3 POKEPL+K#I,3 2 HEXTK 385 IFG=1THEN240 386 IFQ2#1THEN251 390 RM=RM+1 391 IFRM>11THFNH=H+1 RM=0 392 IFH>12THEN250 400 GCT0251 500 FORK=1T02 PRINT"(HOM1"SPC(20),"S 501 FORX=1T010 PUNEH2,108 POKEL2,223 POKEW2, 17 FORJ=17010 NEXTJ 502 POKEW2.0 NEXTX 503 PRINT'[HOM3", SPC(20), "[45PC]" -FO RX=1T0180 NEXTX NEXTK 504 R=R+1 W=0 PRINT"[HOM]", SPC(37), R 505 IFR=3THEMPRINT"[HOMIESPC][RVSON] FANTASTICOLRYSOFFILIUSPCI" GOTOSUZ 586 PRINT"[HOM]PULSA[SPC]/F7/[SPC]PH RAISPUJEMPEZAR. * GOTO509 507 FORX=1TU/5 POKEW2 33 POKEH2,X PO *EL2,100+2*X FORJ*1T010 NEXTJ NEXTX 508 PUKEWZ/0 GOTO700 509 GETS# IFS#=""THEN509 510 PRINT"[HCM][24SPC]".0=1.GOT0380 600 PCKEH2, 8 POREL2, 23 POREM2, 33 FOR XM1TU900 NEXTX PUREM2.0 601 U2=1 RM=MN H=HR W=W+1 683 GL F0388 650 PRINT"(HOM3",SPC(9),"CURREGIDO" 02=0 652 FURX=1TU2000 NEXTK

653 IFW>2THEN10 654 PRINT"(HOMIPULSAESPC]/F7/[SPC]PA RHISPC]EMPEZHR.[2SPC]" 655 GETS# IFS#=""THEN655 657 PRINT"(HOMI[26SPC]" 660 Q=1 GOTO380 700 FURX-1TO1000 NEXIX 701 GOTO150

Es la "Hora" de aprender, pág. 18 VIC-20

8 REM PARH EL RELOJ VIC-20 1 L\$="782678497915800380248020801779 947906784078197822" 2 LH\$="78277871793780258023802080167 97279067818782078221 3 TH\$="+2178+2178-0164-2377-2377-226 6-2178-2178+0164+2377+2377+2266" 4 IP\$="+2178+2178-0164-2377-2377-226 6-2178-2178+0167+2377+2377+2366 5 B\$="777877847807791780498070806480 598036790477947773 6 CO=30720 V=36878 S1=36874 S2=36075 .S3=36876:SC=36879 POKEV,15 10 POKESC, 174 PRINT"[CLR][3CRSRR][RVSON] DIAGRAMHISPOIDEISPOIRELUJERVSOFFI' P RINT"[CRSRD]",SPC(8),"00" PRINT"[65PC] [500MM@15" 15 PRINT"[3SPC]55[SHIFTN][2SPC][2[SPC] [SHIFTM]" PRINT"[4SPC][SHIFTN]11[4SPC] ICSHIFTMICSPC310" PRINT"[SPC350[SHIFTN] [95PC][SHIFTM]" 20 PRINT"[2SPC][COMMM][0[SPC]RELOJ[2SPC] 2[LOMMG]" PRINT"[25PE][CUMMM]", SPC(1 1); "[COMMG]" 25 PRINT"45(COMMM)9(ZSPC)(SHIFTQ)(SHIFT#) [COMM+1055PC][COMMG]]5" PRINT"[25PC] [COMMM][55PC][SHIFT-][55PC][COMMG]" 30 PRINT"[2SPC][COMMM]B[4SPC][SHIFT-] [45PC]4[COMMG]" PRINT"[SHC]40[SHIFTM] [45PC][SHIFT-][45PL][SHIFTN]20" 35 PRINT"[4SPC][SHIFTM]7[2SPC][SHIFTQ] (2SPC)5(SHIFTH)" PRINT"(3SPC)35[SHIFTM] [25PC]6[2SPC][SHIFTN]25" 40 PRINT"[6SPC][5COMMT]":PRINTTAB(7) "3Ø" 45 PRINT"[CRSRD][4SPC]HORA=[SPC]9 30 50 PRINT"[CRSRD]PULSA[SPC]/F7/PARA[SPC] EMPEZARICRSRD10[SPC]'I', [SPC]INSTRUC CIONES.". 55 GETS# IFS##"THEN55 56 IFS\$="1"THEN58 57 G0T0288 58 PRINT"[CLR][4SPC][RVSON][RED][NST RUCCIONESCRYSOFFICELUJ" PRINT"[CRSRD] INTENTALSPOJPARARESPOJELESPOJRELOJ" 65 PRINT"CUANDOESPOJLASESPOJMANILLAS [SPC]".PRINT"[CRSRD]INDIQUEN(SPC]LA[SPC] HOPA. " 70 PRINT"[CRSRD]GANAS[SPC]CON[SPC]3[SPC] ALIERTOS." 75 PRINT"[20RSRB][RVSON]TEULHS[SPL]D ECSPC CONTROLORVEONE 1" PRINT"[CRSRD] 'F1'[SPC]MUEVE[SPC]LA[SPC]MANILLA[6SPC] CHICA" 80 PRINT"[CRSRD] F3/LSPC]MUEVEESPC]L ALSPO1GRANDE 85 PRINT"[CRSRD]/F7/[SPC]PARH[SPC]EL (SPC)RELOJ" 87 PRINT"[20RSRD](PULSA[SPC]UNAISPC] TECLA. > " 90 GETS\$ IFS\$≈""THEN90 150 PRINT"CCLRICCESEDICSPCIERYSONILI

FICULTADESPOIDELISPOJJUEGO" 155 PRINT"[2URSRD](1)[SPC]HORAS" PRI NT"ECRSRDJ(2)ESPCJHORASESPCJYESPCJ1/ 21 SPC THORAS* 160 PRINT"[CRSRD](3)[SPC]INTERVALOS[2SPC] DEC2SPC15C5SPC1CCRSRD1MINUTOS. 165 PRINT"[50RSRD]PULSH[SPC]1,2,0[SPC] 3. " PRINT"[2CKSRD]PULSA[SPC]'F'[SPC] PHRHESPEJFIN. 170 GETD\$ IFD\$=""THEN170 173 IFD##"F"THENPOKEV, @ POKESC, 27-PR INT"ECLR3" END 175 D=VAL(D\$) 180 IFDD3GRDC1THEN170 200 PRINT"[CLR]" R=0 W=0 POKESC, 159 210 PRINT"[4CRSRR][14COMM@]" 211 PRINT"[2CRSRR][RVSON][5HIFTE][RVSOFF] [SHIFTE]]][[4CRSRR]]2[SLRSRR]][[SHIFTM] 212 PRINT"[CRSRR][RVSON][SHIFTE][RVSOFF] [SHIFTE]", SPC(16); "[SHIFTM]" 213 PRINT"CORSREJERVSUNJESPOJERVSOFFJ 10",SPC(15),"2[COMMG]" 214 PRINT"ECRSRRJERVSONJESPCJERVSOFFJ SPC(18): "[COMMG]" PRINT 'CURSER] ERVSON] [SPC][RV5GFF]",SPC(18),"[CUMMG]" PRI NT"[[RSER][RVSUN][\$PC][RVSUFF]"SPC(1 €>,"{cŏMMb]" 215 PRINT"[CRSRR][RVSON][SPC][RVSOFF]
".SPC(18),"[COMMOI" PRINT"[TRSRR][RVSON] [SPC][RVSGFF]"SPC(18),"[COMMG]" 216 PRINT"[CRSRR][RVSON][SPC][KVSOFF] 917CRSRR3[RVSON][2SPC][RVSOFF][7CRSRR] 3000MM63" 217 PRINT"[CRSRR][RVSON][SPC][RVSOFF] [BCRSRR][RVSGN][25PL][RVSGFF][BURSRR] LCOMMG14 210 PRINT"[CRSRR][RVSON][SPC][RVSOFF] ".aFt(18),"[CONNG]" FRINT"[LKS*R][RVSON] [SPC][RVSOFF]",SPC(18),"[COMMG]" PR1 NT"CCRSFRJCRVSUNJCSPLJCRVSUFFJ", SPCK 1877 [LOMMG]* 219 PRINT"CCRSRRJERVSONJESPCJERVSOFFJ SPECIES, "COUMMOD" PRINT"[CRORN][RVSON] [SPUJIRVSOFFI] SHURTS OF TOURMOI 220 PFINT"[CRSRR][RVSON][SPC][RVSOFF] 8" SEL 167, "4LLUMMGJ" PRINT"[CRSRR][RVSON] ESPCIERVOURFI" SPUCIBA, "ECOMMGI" 221 PRINTMECRSREJECOMPWJERVSONJECOMPWJ ERValIFF 17(6UM JAK IDCBLASHA 15(URBRR) (SHIFTN) PRINTTISCRURRILISCOMMITTO 240 HR=INT(RND(1)#12+1) G=0 242 IFD=3THENMH=INT(RND(1)412)-GUT02 4H 243 IFD=2THENMN=6#INT(RND(1)#2) 6010 248 247 MN=0 248 IFMN=WORDN=1THENPRINT"[HUM]",HR, [SPC]", RIGHT# ("0"+RIGHT# (STR#(5#MN 2,12,22 GOTE250 249 PRINT"[HUM]", HR, ".", 5#MN 250 H=1 -RM=0 251 IFRM>2THENPL≃VAL(MID\$(LH\$,(H-1)* 4+1,4>> GOTO286 270 PL=VAL(MID\$(L\$,(H-1)#4+1,4)) 280 PB=VAL(MID*(B\$,4*RM+1,4)) 284 IFRM=0THENII=22 PP=66-GOT0290 285 II=VHL(MID\$(IP\$,(RM-1)*5+1,3)) 286 PP=VAL(MID\$(IP\$,(RH-1)#5+4,2)) 290 POKEPB, 160 POKEPB+CO, 0 300 FORK=1TOS POKEPB+K4II.PP PUKEPB+ K#II+CO, Ø NEXTK JUS POKES1, 240 FORK=1TO10 NEXTK : POKE 51.0 310 IFRM>2THENI=VHL(MID\$(IH\$,(H-1)\$5 +1,3>> P=VHL(MID\$(IH\$,(H-1)#5+4,2)) 60T0330 320 I=VAL(MID\$(IP\$,(H-1)*5+1,3)) P=V AL(MID#(IP#,(H-1)#5+4,2))

330 PUKEPL, 160 POKEPL+CO, 6 340 FORK=1TO3 POKEPL+k#I,P POKEPL+k# I+CO.6 NEXTK 350 FORK=1TOBUG NEXTK .53 IFQ2=1THEN65⊌ 355 GETS\$ IFS\$=""THEN379 357 IF5\$="[F1]"THENRM=RM+12 G0T0379 358 IF5\$="[F3]"THENRM=RM+2-G0T0379 359 IFHR*HANDMN*RMTHEN500 360 GUT 2600 379 POKES2,240 FORK=1T010 NEXTK-POKE 52.0 380 POKEPB, 32 FORK=1TO5 POKEPB+K#11, 32 NEXTR 381 POKEPL,32 FORK=1TO3 POKEPL+k#1,3 2 NEXTK 385 IFG=1THEN240 386 IFQ2=1THEN251 390 RM=RM+1 391 IFRMD11THENH=H+1 RM=0 392 IFH>12THEN250 400 GOTO251 500 FORK=1T02 PRINT"[HOM]"SPC(14),"S 1[45PL]", 8+1 501 FORJ=1T015 FORX=1T010 POKES3,230 NEXTX POKESS, 0 NEXTJ 502 PRINT"[HOM]", SPC(14), "[45PC]":PO KESS, 0 FORX=1T0200 NEXTX NEXTK 503 R=R+1 W=0 PRINT"[HOM1", SPC(20).R 504 IFR=3THENPRINT"[HOM]FANTASTICO![7SPC] " FORX=17050 POKES3,190+X NEXTX POKE \$3,0 GOTO766 505 PRINT"[HUM]PULSA[SPC]F7[SPC]PRRH (SPCJEMPEZAR." 506 GETS\$ IFS\$=""THEN506 510 PRINT"(HOM](225PC1")-Q=1 GOTU380 600 POKES1,190 FURX=1T0400 NEXTX POK ESI,0 Q2=1 RM=MN H=HR W=W+1 603 6010388 650 PRINT"[HOM]",SPC(9),"CURREGIDO" uZ=el 652 FUPA=1102000 NEXTX 653 IFW/2THENIE 654 PRINT"[HOM]PULSA[SPC]F7[SPC]PARH I SPOJEMPEZHR." 655 GETS# 1FS#=""THER655 657 PRINT"[HOM1[22SPC]"; 660 0=1 0010380 700 FOMA=1TL1000 NEATX 701 GUT0150

Bingo, pág. 39

1 POKE53280.5 POKE55281.5 K HRIGHT BOLKET FOR A NATH A FRI 4 141 95B [111], can 4 PRINT*CCLR3(19) RSR() TF11CRSPF3(BLK) N 1114 PART "MO HAC 1943 JADAL (46.134446.4 MM NESTT 5 ยกรปฐ -ผ 16 gir lige sti 20 CLEU FERM PEW PRINT"[CEM][REFT" 30 FORI=W-STOW EUREL,121 POKEI+ (20.1 Par Ne T1 40 FORT=WestZTOH-ISTEP44 POKET:112 Po **kET+8:118 NEXT1** SU FURI=E-1706-415TER4U MORET 11× FO FET+2 112 NE TI 60 FURT=F+39TUF+41 PURET 120 NEXTT 70 FORI=1109 FORI=1TO10 W=W+3 PONEW,

102 POFFW+1-107 NEXTT W=W+50 NEXTI Se Purioli, or Volume NE II MHARAN II = 1 85 PRINT"[22] RSRIGHUMEPOLSPOIANTER TO 87 PMINITABOLEOUTEN REMITERUL SHRESPOLERVSUNI LICRYSOFF 31 SPC TPARHESPC TUTNEH" 88 PRINTHBOTHS "PULSHRISPOTERYSONTRIRVSOFF] के न कीए सीज़िंग पि) Hamai पर j 90 FOR LEHTUR POLETICE, 6-POKET, 81 NEXT 95 PONEC+24,15 PONEC+6,128 PONEC+5,1 235 100 IFTT AN HENE OF 110 NU#INT(PHID TI)#MHX)+1 W=1032 GOS UB4MU 100 FURK#1T02 14и 1- БТ-КИБСЕТ Ф:К-н+1 / +н -11(f Fed) 11+10# B-H+100+H 1504 1em 1e _= 11eem15m 1 m = 1el em 1 +c • B=e+1 +e THO THE WALL PROPERTY. 1 of L=-1 GUS(BBBH) AND LET XI=D-40 GUSUBRNO GUSUBRNO IF P=1 THENK =2 21M HEXTE 215 IFP=11HFN6 NU 217 PIREC+4, 33 2/0 FUNT=H+ TO1/885TEP40 PC#EI+C.5.P CKET, 32 POKET+40+0,6 PUKET+49,81 225 PUREU+1,1750 PUREC,1720UHND,55 N EXTI PURFOHALIS 238 GOSURBON IFFET THENS KA 444 PIOEFFI,, PITE FT+4 POFF+1+ ,
Z FITEFFT 11+4 F P 11 is the T 250 PUKEPO+C.1 POKEPO.FI+48 POKEPO+1 +0.1 POKEPO+1.00+48 260 PUKEU+C:1 POKEG:FI+48 PUKEG+1+C: 270 GOSUB8ии [FP=1THFN630] 280 PUKEF+C.5 PUKEF-32 PIKEF+1+C.5 P UFEF+1, REPORFE+LINE PURFE, 124 A PIFHE + PERFERS 5 in alborn (FM=) THENESE ह का नाम्माम्बद्धाः । जिल्ला 31141 FIRE 1 +4 1 1 12-04 13-44 10 FORI=1TU5 3 0 IFI THEFAT =4M 3.0 IFT STHEUL - 40 40 1 - 1+400 + I-10 FIR POTEBAL S ALTER 4" FIRE Z+C. 5 PC RE / SE POKEYS+C.5 POKEXS, 32 + + + - (m - t+ | t+, p 5 f 365 NT=50+L+(12*I-42) PEREC+1.NT POR , BILL HALL 255 370 PUKEX2+C. A POKEX2-81 POKEX3+C.6 PUNEX3,81 PUNED+L+4MAL,6 PUNED+40*L, HI. Sent NE 11 MCFF +4 M METURN 465 - -167 NO 18 C WYNO-FI***1U** I Hens+W=DH 416 1500 STHERPOPPER BY GOTHARD 4 S F 190 SM 4 a RETURN 440 (Figuration (Tel-1446)) 450 FORT=NUTOMHX-1 V(T)=V(T+1) NEXTT 460 467 BH -1 I= [T+1 वास हम भार 4.5 h Elf H 4 in IFEL=1HHD: \$="L"THENPPINT": THEHEAPC) भारता है। विकास समित वार्ग के 490 [FBR=1HNDS#="B"THENPFINT"RINGUISPO] YHLE I HICE COLUMN SOU IFS#="L"THENLL=1 PPINT"LINEHLSPU] EURRELTHISPIDIS/NO?" GOSURS60 GOT057 510 TESS="R"THENERS PRINT"BINGUISPOT DERECTOR PUBLICATION OF BUILDING

520 PRINT"ERPORESPONALISMO IPULSAR" P RINITAB(17) "PULSELSPE IDELSPE INCEVULSPE1 [RVSON]L(RVSOFF][SPC 10[SPC][RVSON]B[RVSOFF] 525 GETS# IFS#=""THEN5/5 5 80 GOSHB1170 54m GOTO 48m 550 PRINTTABOLZ⊃"CONTENUMMOS" FOPK≖1 TOBBURD NEXTH GOSUBITZA GOTO624 SEE GETSS IF-S=""THENSELL SKS IFFE IS THINE OF DINT THEN SER SIST PET IPN 570 IPST: 'S"THEROU CRII'S PRINI"LINE HE-PUBLISHED THE PRINTING OF THERMS TOPEN She IF IF HOTHERICEN TO RILLA PRINT "LINEAT SPOINGESPO JOHRPHUTA" GOTASSA 590 IFS\$="N"THENBB=0 GUSUR1170 PPINT "BINGO[SPC]NOCSPLICORRECTO" GOTOSHO 60и IFS\$="S"THENGO-UR117И PRINT"BING M': PUTTURRECTO" PRINTTHEOT'S "GENE" OR "PTS" P=1 61H PRINTINBO17/1961 to TUBLE 1 PESTIN E 911 TEN II FORTALTHAME NEATT 6. U PETURN 5 B FRINT (CLRITTR ALT WITH I FELL OF THE 640 GFTS# IFS#=""THEN640 650 IFS\$="S"THENPRINT"[FRSPDIESCSPC1 LOCSPOINE FORCSPLIDUECSPOIPUETESCSPOIL HHCER" FORT=1TO3000 NEXTT GOTO10 660 IESSO "N"THENERA FIN PERMITTER TO FITTE SP. TO SPECI PTERDES CSPC INDITISH ENTI REG FRINT"LOURD"THE 18 FT REFTERV SOND BINGOTPYSOFFI" PRINT" (20 PSPDI" INPUT "CARTUNESC SPC IVENULTIONS" HA PRINT 690 INPUT"PRECIDESPONDEESPEN HUMLSPCT INTHA 94 NOTHEL THE INPUTION OF IPHER SECTION INCH SECTION **EINT** 710 INPUT"XCSPC IPARACSPC TBINGO", CC-P FIRIT 1 10 TESS+CCVIDIOTHENZION 73M IESS+CC<1MOTHENGL=INTCHREEP+55/1 gg bb-1111 HHΦF+ + 1 1 1 ho 740 GL = INT (AH+PP+55/100) GB=HA+PP+GL 750 PRINT"[20RSPTOPPEMIGESPC]ALSPC] HESPELL THEREIN THES 760 PRINT"[JURSRUPHEMIOL/SPC 10L1/SPC] BINGO"GB"PIS' THE PRINT [+ R-RIDE | EL + COUNTRE CEL] TECUMESPOJPHPHESPOJEMPHPHPH 780 GETS\$ IFS\$=""THEN780 290 PETURN 800 GETS\$ [FS\$<>""THENGOSU8475 **610 RETURN** 828 POKEC+24,15 POKEC+6,8 POKEC+5,12 8 POREC+3.99 Z=3898 822 READM1 . IFM1 = 0THENPOKEC+4 . 0 . RETUR H24 REINTOMIZIO CONCMITIURKONI 826 POKEL +1,0/256 PUKEC . QHND/55 828 POKEC+4, 0 . POKEC+4, 65 FOR 1=1 TO 4 M # (R+1) NEXTI 830 GOTOB22 832 DRTR26, 6, 25, 5, 2, 2, 2, 5, 6, 7, 26, 6, 2 5.5,2,2,2,64,2 2,5,4,4,5,2 2,4,5 6,7 834 DATA15,6,24,5,2,2,2,5,6,7,26,6,2 5 5 272 2 9470 880 PRINT"[CLR]" H=1027 HH=1427 890 FORT=HTOHHSTEP40 POKET,42 POKET+ 8,42 POKET+11,4 P &FT+15 42 NE VII 900 FORI=H+59TOHH-21STEP40 POKET,42. POKET+8,42 POKET+13,42 NEXTI 910 FORT=H+1TOH+3 POKET,42-POKET+200 ,42 POKET+400,42 NEXTI 920 POKEH+4,42 POKEH+164,42:POKEH+24

4,42 PUKEHH+4,42 930 FORI=H+45T0H+125STEP40 POKEI,42 POKEI+240,42 NEXTI 940 FORI=H+94TOH+334STEP40 POKEI:42: NEXTI 950 FORI=H+20TOH+23 POKEI:42 POKEI+8 42 POKET+400-42 POKET+408-42 NEXTI 960 POKEH+12:42-POKEH+53:42-POKEH+37 5,42 970 FORI=H+30210H+304 POKET, 42 NEXTI 980 POKEH+64,42 POKEH+344,42 POKEH+3 84.42 990 PRINT"[16CRSRD][6CPSRR]PORTSPOJR HFAEL[SPC]MUNYOZ...".FOR[=1T08000:NE 1000 RETURN 1130 PUKE781,18 POKE782,17-PUKE783,0 **ካ**ያጜይ56 44 1140 RETURN 1150 PRINT"[22SPC]":PRINTTAB(17)"[22SPC] 1155 PRINTTHB(17)"[225PC]" 1160 RETURN 1170 GUSHB1130 1180 GOSUBITSA 1190 GOSUBI13M 2000 RETURN

Halterofilia, pág. 41

10 PURE52,24 PORE56,24 FER 20 POINE 36869, 254 22 FORT=6144TU6668-POKE1,PEFK(26624+ I) NEXTI D. PRINT TOUR HOAD ROAD I LUUD SER HERVSON I dH 1 "WYDERCYAJIANENTI"THING 25 26 DUSUB 3000 40 PMINT*(3CRSPD163CMSRR]##HHI TEMOFT L IHe#" 41 PRINT"[40P5RTI][60RSPR](19/V/84)" 42 FORT=1T04MMENERT POFE # \$2+.11 43 PRINT"[CERT" POKE 36869, 240 45 PRINT"QUANDOLSPC1016ASTSPCIELTSPC1 PRIMER" : PRINT"PITIDO : PULSALSPO ILALSPOI TECLH" 46 PPINI "TI R-KLILSI MURKIZI MARKIERECH HI, " FRINT 1 = 15 47 PRINT"ELESPE INHYURESPE INUNERELSPE I TIE 1 PRINTELOPSKI I ZELKSENECTRUSTBUR" PRINT"[CRSPD]HHSTHESPC 100E" 48 FRING FRIND" SUBBLE POSEL (SPC 2/2/5PC) PILLIDIA " 49 PRINT"[20PSRT][RUPSRPT[PVS0N]PULS ACSPC TUNHESPECTECT HERVSOFFC" FOUNDET # TEX##PRIMENTAL 58 PUKE36869.240 PUKE36879.145 59 PRINT"CCLPD" PRINT"L3CRSRLDCCRSRR1 INTENTHESPELICEVANTAR", L PRINT'(BUR-RR) KILOS" 60 PRINT"[7CRSRD1E6/RSRR]HTENCION!!" 65 FORT=ITU4999 NEXET 67 GETS# 1FS#<>""THEN67

70 POKE36878.15 POKE36876,215 PRINT*

82 PRINT"C2CPSPRITESCRSRRILLARTOLL" P.

75 FURT≃110500 NEXTT POKE36878-0

78 GETX# IFX#="[LRSWR]"THENP=P+1

CLE MATERIAL MINERAL TAN STATE

77 H=1ki-TI\$="@brokko"

79 IFTIZEBUHTHENSZ

SØ GUTOZE

UKE36878.15 PUKE36876.215 PUKE36879. 28 83 FORT=110500 NEXTT POKE36878.0 84 IFI >PTHEN1000 85 PUKE36878,15 FURE=13070254 POKE36 876.L 86 FORM=1T040-NEXTM-NEXTL-POKE36879, 143 67 PRINT"[CLR1" PRINT"[BURSED]TETSPC] HACSPOLICISTADO[SPI JEERO... 100 PUKE52,24 PUKE56,24 CLR POKERU86 9.254 110 FORI=6144T06868 POKET, PEFK (26624 +I > HEXTI 115 PPINT"[CLR][4CKSRD][10CRSRR][RVSDN] (FH* 117 PRINT"[10CRSPR1[RVSONTYW" 120 GOSURSoun 140 PRINT"[3CKSRID[CRSPR]] | LOUSPO][00 NSEGUISTEL # 149 FORG#1104000 NEXTO POKE 45869,240 Pull Enbar Yorks 15W PRINT"[CLR]" - PRINT"[3CRSPD]CUHNT UNISPORATEON, SPONGLIFFE ST INPOTT[URSKD] HROKH! 152 POKESH879-57-60TUSH 1000 PRINT"[CLR1" PRINT"[3CRSPD][4CRSPR] LOCSPUTSTENTO, ESPOTSULOS 1010 PRINT"[CRSRD]]ENTHS[SPC]FUER?H[SPC] PHEH! PRINT![[FURG]: EVHN]HR "P PRINT "kILOS 1858 TRPOTHOTROUSELTIBEOUSS, NO 5-Y# 1H40 [FYS="S"THENDUTHAZ 1956 STOP 3000 FURI≖716af07183 REHDX PUKEL X N E 13 DOUBLE FOR 1=2344102359 REHDV PORETVK N EATI SHEEL RETURN SUBUL FORT=7168TO/183 REALS POKET/S N ESTI 5010 DHTH48/51/63/59/57/15/15/7 SOLU BHIR12 284 252 720,156,740,740, 50 O FORI=7344TU7359 REHDX PURELXX N 5м4и ДНТH7,7,7,7,3,3,3,7,7 5050 DHTH2247224 7247274719271927192 144 SUBU RETURN

El Pistolero, pág. 44

2 501-1083-94 4 Infinition of 4 1 4 in on 5 Infined no 2n 4 4d 4 of no 6 Infinity/Inf 155 165 55 474 4 7 DHTH24-60-102-102-231-195-195-195 8 DHTA195.195.255.255.66.66.66.66 ч Дигии и да, аи и и 10 Днгн24,24,24,24,24,24,60.6и 11 PUKE52:28 POKE56:28 CLP FOR1=7168 TUZION+ZAR-I PEHDA MORETAX NEXT PRIN T"[FER]" HP\$(8) 13 РОКЕЗ6869-255 Т1#="йиминии" 15 FORT=1TO12 C=INT(PNU(1)*19)*INT(R ND. 10 #21 ##22 POFE 2726#8 (2) POFE 88446# C.O NEXT 20 T1=8164 J2=8185 B1=6 B2=6 CP=3072 Ø 3-150 S=25 (1-0 (2=0 FUR1-1TU5) (1)≠7822+66♦1 NEXT 30 PRINT"[HUM1[RVSQN]MUFRTUS"M1"[HON] [120RSRR] MUERTOS "MZ

31 PRINT"CHUMICRYSÖNIECR-RÐIL9CRSPRI

* VALENTER TER RESPONDENCE TO BETWEEN \$,255 GUSUB1499 35 POKEJI.Ø POKEJ2.1 POKEJ1+CP.Ø POK ET +(P.VEP RETI+S & PORFIT-S O PORET 2+5.0 POKEJ2-5.0 36 FORT#1TOS PORECCID:4 PORECCID-S:3 PUREI PHO(1) . W PURECH+U(1)-5 @ FOREC 37 ((1)=((1)-22 IFE(1)=7734THENPOKEC (1).(POPEC(1)+22.0 (1)=8174 38 NEXII 39 K1=PEEK(653) K/#PFEK(197) 40 IFK1=IANDC 1=A THENC1=1 - IFB I COTHENB 1=0 41 IFK2=31HNDC2=0THENC2=1 IFB2<0THEN B2=M 50 IFK1=2THENJ1=J1+22 IF 11=8186THENJ 1=8164 55 1FK1=4THENJ1=J1-22 IFJ1=7702THENJ 1=7724 60 IFK2=X3THENJZ=12+X2 IFJ2=8XU7THEN J2=8185 65 IF: 2=54THEN12=12-22 IF JZ=7723THEN [J= 1/45 SECTION OF A SECTION OF THE SECTION 120 IFBI=WHNDB2=WTHENPRINT"[CLR]" GO T015. 130 001039 140 PPINT"[HOM][[CRSP]]]", IFB1=NTHENP MIRT 1875CH. FEST " (150) T '0 145 FORT=110B1 PPINT"F", NEXTT FORT= 11 +- EL PRINT' (R. SHIL-P JERV-OFF) HEATI ISB PRINT"[HOMITCPSRDICIALPSPRI", IF B2=OTHENPRINT"(RVSON)[65PC]" F2=0 FF TUPN. 155 IF 6-B/=0THEN157 156 FORTHITUS-B2 PRINT"[RVSON][SPC][RVSOFF] ", NEXTI 157 FORT=1TUB2 PRINT"F", NEXTI RETUR 160 B1=B1-1 D1=J1+1 FORC1=1T020 H1=D 1+01 165 IFPEHK(A1)=1THENM1=M1+1 170 IFFEEK (A1) = 20RPFEK (A1) = 30RPFEK (A 1) = 4 THE NEUKEHI-1 II GUTCICS 175 PRIKERI, S PURECHHI, 0 PORERI-1.0 189 NEXTUI 185 Fl=M PONEHLO RETHEN 190 B2=B2-1 D2=J2-1 F0PC2=1T020 H2=D 2-0% 195 IFPFEK (H2 /=0THENM2=M2+1 ZHU IFPEEK (AZ /= 20RPFEK (A2) = RHRPEEK (A 2)=4THENPOKEH2+1/0 0070215 205 POKEH2,5 POKECP+A2,0 POKEA2+1.0 210 NEXTC2 215 POKEHZ-O CZ=M RETUPN SON PRINT (HIMITAL MERBITMES HITTER RENR) DIRECTOR OF A LAND CO. L. 301 GETH# IFH#="5"THENRUN 402 IFASCO "N" THENSHI 310 PRINT"[ULR] " HRECED PURE SHORE + 24 B END 320 POKE36869,242 POKE36879,75 PPINT CHR# 37 ELER. [H. 5 HOLD BLD 114-PT1[SHIFTL] CSHIFTOICSHIFTSICSPORT HERIETPICSHIFTEI CHIEFSICHTETTIC HEFTGICSHEET JCHIEFET [SM[KIR][HIRTUILSH(FIS](45P] 330 PRINT"(SPC)(SHIFT)) NEGDESPC 1PARH [SP. IDesCaP | I to. FPIRI'[[MSRD][EvenN] [SPL]LEHIETUJ(SHIETI)U-HIET 11-H'ET]] (SHIFTOICSMIFTNIESPOICHVSOFF) [SPC](SHIFTO) HRUHTEL EN IME SPICEN SONTRARIUS P. 1 SIISP01" 340 PRINTSPCC10 , "PUEDE 5." 350 PRINT"EPOPSRBIESHIFTFILMETUN | F] C-HIETJIOS, CSECTICSECTE-HIETJIOSCOPET

360 PRINT"[CRSPD][SPC][SHIFTH]RRIBH(2SPC]]
[RV-ON][SPC][RV-ON]
[2.P.]1(SPC][RV-ORF]]
365 PRINT"[SPC][SHIFTH]BHIOLKSPC][RV-ON]
CONM, [SPC][RV-SOFF][2SPC][RV-SON][RV-SON]
[RV-SOFF]]
370 PRINT"[SPC][SHIFT]][SPRPOCSPC][RV-SON]
SHIFT(SPC_[RV-SOFF][SHIFTP]ULSHC-SPC][RV-SON]
[RV-SOFF]]
375 PRINT"[CRSPD][SHIFTP]ULSHC-SPC][RV-SON]
[SHIFT-][HIFTH][SHIFTP]ULSHC-SPC][RV-SON]
[SHIFT-][HIFTH][SHIFTP][SHIFT][SHIFT]]
SHIFTO-[HIFTH][SHIFTP][SHIFT][SHIFT]]
SHIFTO-[HIFTH][SHIFTP][SHIFT]]
SHIFTO-[HIFTH][SHIFTP][SHIFT]]
SHIFTO-[HIFTH][SHIFTP][SHIFT]]
SHIFTO-[HIFTH][SHIFTP][SHIFT]]
SHIFTO-[HIFTH][SHIFTP][SHIFTP][SHIFTD]

Master Mind, pág. 45

10 PEMPAMMSTER MINDO 15 DIM 10 14) 28 POFE53281 - 1 30 PRINT"[(IR]" 35 T14="MMMMMH" 40 PMINTTHBC147"[REDIERY-ON]MHSTER[SPC] MINDIRVSUFFI" 60 PRINT CLUBERT' FRINT IF OUF I REAL TRADOCSPC THOMOSPOOTS, CSPC TSHLHS & P ULSHIPE I HAT POITE CH 70 DET MA IF MARTHURITO ZE BU PRINT COLFTCE SHITL SPOTELTSPOTOR DENHIURCSPC IFSCHOECSPC TUNCSPC INCMEPO CSPC TALL SPC THZHP. WE MEN NEO BEN PESTURE 100 PRINT"COORSEDIDECSPO ICCHANTOSC SPC 1 TERMINOSESPOIDESEACSPOIEUTSPOINOMERO (1-6) 105 INPUT A 107 IF ACT OF HOS THEN BU 109 D\$="(45PL)" FOR 1=1 TO 8 D\$=D\$+" IBLK HISPLI" NEXT FOR I=1 TO 49-24A D \$=II\$+"[SPC]" NEXT 110 PRINT"[20PSPDITIESFHESPE 100FESPC1 SELSPO JEUELINN, SPOAREPETART - FATERMIN ûS1 120 INPUT"(S/N)", R\$ 125 IF B\$CO"S" FAID R\$CO"N" THEN 124 130 PRINT PRINT PRINT PULSETSPE TOTHE DUTER (- PC 1 TECCH! 135 GET C\$ IF C##" UFFO 135 137 IF R\$="S" THEN GOSURIONO GOTO140 1 чя бизия2ими 140 GOSUB BAN PRINT"(CLR)" GOSUBSNOA 150 M=M+1 60SUB 7000 168 FOR 141 TO M 170 GET C\$ IF C\$="" THEN 170 188 IF C\$ "1" OP C\$>"9" THEN 11" 198 (*1)=vHL (C\$) PRINT(&*[AF]) PUM NEWT 265 BUSINESS 214 GUSUR JAMA IF N=A THEN 480 220 PRINTN'O ROH PRINT 225 IF MOR THEN PRINT"[HOM]" GOSUBSM WO PRINT"[16CREPD]". 230 GRTU150 300 PRINT"[CLR][2CRSPD][KCRSRR][FED] "; FOR I=1 TO A PRINT LEFT#(STR#:B(1 11.21, NEXT 340 PRINT PRINT PRINT "HECHOESPOJENISPO] ",M, "INTENTOS" GOTGIGNUN 345 605UR 8000

360 GET H\$ IF H\$=""60TO 360

365 IF H\$="S" (00TO 7M RZW END 510 INFUL (\$(I) 5.20 NE, T 1 5 O PETUMN 888 PORE 54, 46, 9 FOR 7=1 10 7 PEHD L KZ + NF (1 81M FORKSTIONA REHD TOXO NEXT FOR EST TUZ PUKE 54275/L(J) FOR 2=1 TO B(2#1 -10 NEXT 8 W PUKE54275.9 FOR 7=1 TO D(241) NE 830 NEMT J 840 DATH 195-195-195-199-195-195-201 FEB. EM. BM. BM. BM HEAL RETURN 1000 FOR 1=1 TO A BCD=INTCRHDC03#93 +1 HEAT I RETURN ZMBW FUR L=1 TO H ZUIU B(I)=INT(RNb(Ø)#9)+1 2020 FOR Z=0 TO I-1 IF B(I) = B(_) THE h 2010 2030 NEXT Z NEXT I 2040 RETURN SUMB RED NEW FOR THE TO H IF COLDER I) THEN N=N+1 BUHS NEXT JULU IF NEW THEN RETURN 3048 FUR IM1 TO A BRISH FOR JET TO A IF CCT) # KCD THEN R=B+1 3040 NEXT I NEXT I BER-N RETURN SUNU PRINT"THUMTCREKIM#MHERTUC65PF 1H #HER LIDET 2SPC TORE DO " D\$" CBLOT" RETORN 6000 FOR I=1 TO 2 PORE 54296.3 PORE F4273, 215 6010 FOR Z≈1 TO 150 NEXT Z 6020 POKE 54273.0 6030 FUR Z≈1 TO 150 NEXT Z EWAU NEXT I 6050 RETURN 7000 M#=STR#(M) PRINT MID#(N#,2,2)" "THB(4), RETURN 8000 PEKE 54296, 3: FOR I=1 TO 3 PAK Flot 76 25 BH10 FUR Z=1 TO 100 NEXT Z SHEB PUFF54/74 W 8030 FOR 2≤1 TO 100 NEXT Z BUAN NEXT I 8050 POKE54/95,220 FOR Z=15 TU W STE 886И PH E54296,7 FOR M=1 TO 100 NEX BUTU NEAT . 8080 POKE 54245-0 POKE54296-0 RETURN SETTING SHOW FORTHLION PRINTECTS, CCTS NEXT THURSE FRINT "HHST POTTER DEBUT TR PRI" MIDSCTIS, 3, 2) "COURSER INTRIDIS" lines [#=cMID#CTI#.5 TORRES PRINT IN "CATANA SEGURIUS" SERVICE SECURIO SECU ALSECTION FOR DICKA SIGNATE 64901 PRINT"[CRERIGI" (((H+5H4b)+) H#1890 >= CER#1980 + C[0+(7) >) " [1 RSPR][RVS0N] PUNTOSCRYSOFFI" BBBBB RRECTORPSMARAM HAIMMODELCHAIMM 1+(I)+(Z))) 6,905 IF PRINCIPLE PRINCIPLES ON INDE ALTPELTERIOR DURANTER SU ENTINEULT & BRENE. 14** (4** [[] 4*] 4] 63945 PRINTTERV ON IRPORPIGE CHEEDING

b 4999 PMINT PRINT OUTENET SET THINKET BY TOTAL BUILDING BUILDI

и



METEDURAS DE PATA

En nuestra página 53 del nº 7, se repite "del C-64, pero es difí" cuando debería decir "con un solo drave".

Lineas que faltan en "Nervios" de la sección Videocasino de nuestro número anterior.

400 IFFR=ROGOTO630 410 IFFR(1)>500ORCH(2)>500GOTO730

Errores corregidos del programa "auto" de la sección magia (C-64) del nº 7.

10 X=49152 20 READY IFY -- 1THEN40 39 POKEX, Y X=X+1.Z=Z+Y GOT020 40 IFZC>12374THENPRINT"ERROR[SPC]EN[SPC] LASESPEJSENTENCIASESPEJDATA" END REM SI Z ES DIFERENTE ... 50 SYS49152 NEW 100 DATA169,90,133,251,169,8,133,252 ,169,19,141,2,3,169,192,141,3,3,96,3 2,25 110 DATA192,76,134,164,24,169,10,101 ,251,133,251,144,2,230,252,165,251,1 33,99 120 DATA165,252,133,98,162,144,56,32 .73,188,32,221,189,162,0,189,1,1,248 19.32 130 DATA210,255,157,0,2,232,208,242 32,18,225,201,13,240,3,76,105,165,56 .165140 DATA251,233,20,176,2,198,252,169

.131.141.2.3.169.164.141.3.3.76.118.

Errores de "Aventuras en el desierto" de nuestro nº 4 (VIC-20).

FEADY.

165,-1

122 POKE38622, WC POKE38624, WC

READY.

500 POKE36869,255 POKE36867,25 POKE36879,31

READY.





Micro Sistemas

y le llevaremos a su casa un APPLE II c

Suscribase a MicroSistemas antes del 30 de noviembre próximo y participará en el sorteo de un magnifico ordenador personal APPLE II c.

Para los que ya estén suscritos también tendrán opción de conseguir el APPLE II c. Si desea conocer su número en el

sorteo llámenos al teléfono 231 23 95. Habrá también otros premios para nuestros suscriptores que daremos oportunamente a conocer. El sorteo tendrá lugar en nuestro stand del SIMO.





BOLETIN DE SUSCRIPCION A MICROSISTEMAS

Nombre y Apellidos

Paesto

Compañía

Actividad Empresa

Dirección

... D. P. Provincia

Suscripción anual 2.950 ptas

Deseo iniciar la suscripción con el próximo número

Adjunto cheque por 2.950 ptas

☐ Reembolso más gastos del mismo al recibir el primer número de la suscripción (Enviar a Gravina, 13-1º Izda. Madrid-4 - Teléf· 231 23 88/95)

de las siguientes instrucciones.

10. POKES para protección de juego de caracteres,

20-60. Traslado de los punteros al juego de caracteres para posterior utilización de ellos.

70. Pixel de orientación del comecocos hacia la derecha

80. Pixel de orientación del comecocos con la boca cerrada

90. Pixel de orientación del comecocos hacia la izquierda

100 Pixel de orientación del comecocos hacia abajo

110. Pixel de orientación del comecocos hacia arriba

120. Pixel de creación de muros del laberinto

130. Pixel de creación de bombas

140. Pixel de creación de monstruos

150. Pixel de creación de espacios en blanco

160. Pixel de creación de bombas

170. Pixel de creación de puntitos de puntos

200 Data final

210-280. Instrucciones.

290. Carga automática de la 2ª parte y vuelta a la normalidad de caracteres.

2) Es la parte donde se desarrolla todo el juego:

430. Indica los bytes libres de que se dispone

450-470. Variables principales. 460=variables para utilización del joystick

490. Subrutina de presentación

500. Conversión al juego de caracteres y función de teclas repetitivas

510. Color de fondo-pantalla: blanco

550-640. Creación del laberinto y de los puntitos

700. Pregunta al teclado.

660. Bombas iniciales. Van aumentando.

720. Si consigues 2,200 puntos y te comes aquel signo, pasas de pantalla. Luego debes conseguir otros 2 200 puntos y así su-

I REMA PRIMERA PARTE A 5 BENFLHEZPINTH PHOF# THE REPLACE PORESS OR LO PURIEZIBATOZ, 49 PONET, PEEK (T+2560) MIT NEW T sit state amend , the AU REMUM TEMA-LIMENTAM 50 FURT=0107 REAUB.PUKE7168+H48+T.B NEXT 70 DATH50 >60-126-118-126-112-112-12 ь ын BU DHTH51 760-126-118-126-126-126-12 6.60 90 DATH52 -60-126-118 1.6-14-14-126-100 DHTH53 ,126,255,255,255,247,207, 207,78 110 DATA54 .78.20/ 201 247.215 255 2 55 126 1.0 DHTH55 (25%,255),55,255,255,255, 255,255 130 DHIMSH RESIGNATE SELECT 1.4.124 140 DATA57 ,28,62,127,93,127,127,127 ปริยั โฟโติริส เซเซเซเซ ท ซ ท.ย 160 DHTH59 -16 16,56,56 56 56 124.12 170 DHTHEO .0.0.0.16.0.0.0.0 ZNU DHIA-1 210 PRINT"[CLR]" 220 PRINT"ERVSONJEREDI****LHBYRINTHESPCI PH" | *** (B, 11) " 2 34 PRINTYO REROTTACHSMRILRVE MILBLED = F PC EDERECHHERYSUFF 1 4M PRINT"[CMSMD][4CPSPR][MVSON] [SPC] IZHRIERDHERVSOFFIR,4 250 PRINT"[(RSRD][40R5RK][RY50N]5PG[25PL] HBH JOTRVSOFF 1. ".5

730. Variables para la utilización de joystick,

740. Control de packman extra

800-850. Controla la posición de packman y sus choques contra los muros.

900-970. Control de contelleo de packman y sus movimientos

980-995. Indica los puntos y cronometra el tiempo de aparición de los monstruos. Cada vez aparecen un seg menos. 995-control de retardo de aparición.

2000. Subrutina de sonido de choque contra un muro

2500. Subrutina de sonido cuando come un puntito y contador

de puntos

3000. Subrutina de aparición de monstruos, hasta la línea 4.060, junto con la aparición de bonus extra —4000, 4020— y de bombas —4060—. Si la salida de un monstruo coincide con la posición de una bomba o de packman pasa a la subrutina 4500

4500-4580. Subrutina de sonido de destrucción de un packman. Si quedan 0 packman va a la línea 6000

4995. Primera subrutina de cambio de pantalla

5000-5120. Subrutina de presentación y de cambio de pantalla CRLDITS

6000-6090. Subrutina de fin de juego

6500. Subrutina de sonido de packman extra, —son cada 7500 P—. Abarca hasta la linea 6540

7000-7040. Subrutina de sonido cuando comes un bonus extra

Como pueden apreciar, este programa está realizado con graficos definidos por mí, por lo que quiero felicitar sinceramente al equipo de COMMODORE WORLD por haber puesto un artículo sobre la falta de resolución en el VIC-20 Sin embargo quiero aportar "algo", si se puede llamar así, aportando un tema a tratar en la revista, y valga la redundancia; la utilización del cartucho del SUPER EXPANDER en el VIC.



REGISTER LATELY CONTINENTAL, S.A. Balmes-297, pral. 2º A BARCELONA-6 Teléf. (93) 200-18-99 Información: Sr. FFRRFR

NUESTRA EMPRESA AL SERVICIO DE TODOS

DEPARTAMENTO COMERCIAL:

- Microprocesadores y ordenadores de gestion (gama Commodore) para el particular y I mpresa
- Programas educativos, de juegos y de diferentes aplicaciones (Gestion, contabilidad)

DEPARTAMENTO DE ESTUDIOS:

- Cursos Programación Basic y Cobol
- Cursillos de grabación
- Prácticas con ordenadores y micro en la propia Empresa

260 PRINT"LOPSRDJE4CRSRRJERVSONJ@ESPC1
HRPIBHERVSOFF] ".6
J7M PRINT"(ROWSRDJE SEFERE L. N' . NJPUL
SHRESE JUNACHPOITE CHERMSOFF]"
ZBO WELSE JENSE"/THEN M
, AN PRINT"[...PJE & FID. F JCHPOHNDOESPC]
HETHESE JRE GEORG SEFEJ LORECH.FDJ
' POREBSJ.131 PORESPS 1 PORESRBS 24
U

SEGUNDA PARTE 430 PRINTERECOD, "BYTE SUSPUJERLE" FOR T#1TU1000 NEXT REC U 450 SC#U HU-8 LI=3 TY=1 Lk=/*and (k=1 460 PORE37151-0 PH=37151 PR=17152 RR 47(0 41 Y413 No. 4)/ U TEBATOMOGRAP U Narabe burth W 368's asalenie ने भाग का देव_{ें} किया कर है। 500 FORE JABER 255 PORE650-128 510 POREVO+1-25 NEW PRINTMICERICHOMOLORSKIDIEBUK 17777 590 PETRT"? 7 3 x x 1/2 232 2 25/2/25 - 42 272 - 3 4/9 3/2 600 PEINT' 2 2 2 2 7 7 7 7 7 77 77 71 7 38 78 1111 72 74 5 6 7 78 6 8 < 77" 620 PRINT#7CCC74 177777 7 74 1/5 7777575CSC 7775777C757 SCC27 77745C 17777 7 7 7 36 71 EN RUMBETTOTY FREINTIENE (19418 +44 POREL HARD SA PURE SANDRAHOUZ NEXT 700 GETH# 710 POKE 77. % 0 720 1F50 = 1,0 MB+TORNDPEEK (QM+X+Y#22)= ативиот съв отпеки 7 & H-FEEK (PH) PUKERB, 127 B=PEEK (PB) PHIERBI255 740 IFSC =UNTHEROUSUBREME EDA (FICHHALM FAMIHENVAY-1 MIRSA) AND TELLER COMEXAGEN #250 REFERROUNDS Y#IZ 7=59THENY=Y+1 GOSUB2000 810 IFCHHND82=UTHENY=Y+1 MU=53 815 IFFEEK (QW+X+Y#22)=550RPLEK (QW+X+ Y#2, D=59THENY=Y-1_605UB2088 as in the Hamble conflicts as 1 111=52 SUB-TREEFE - NEW TANKS THE DE REFERREDHENSE V#22/=5/THENZEX+1 - N. IB/SERRE 8 40 IF FHRID, 28 = 01HEN = +1 MO=50 RICH IEPEEL (UM+, +Y#ZZ =550) FEEK (UM+X+ Y#22145+THEN 4) -1 (in 118, 60) a5w IEPEEK+UN+X+Y#22x=601HENGOSUBa5u 860 IPPEEN CON+X+Y+227=56THENGUSUB700 910 POREON+x+7+x MO 92M PORE SHINE +++ . 6 958 FORTATIONAL REST 960 PUKEUW+X+Y#22.51 970 FORT=1T0100 NEXT 980 PRINT"LHOMICRYSUNILGRNISCORE ",S C;"[SPC]TINE ", VAL (TI\$)

990 IFVAL(11\$>=30-HTHENGUSUBBRUU H≠H 945 IEHD=15THENHAU THINK FLAT AFAFFA. SU 1589 GUTH MA 2000 POKEYO,15 POKES3,240 POKES3 0 R FLUEN 2580 POKEYO, 15 POKEU3-1, 255 POKE53-1 U · = < + LU FETURN SUMM FURTHER TUSHED JOHN BOY TOTAL POLICE (#41) CH44 30 NOTEPHER OF MARKETS TO THE MESSAGE 30 JA TEPEEK (QUIHAR) =590RPEEK (QUIHAH) =5 1 THE NASSANIA 3040 PUNEUMHHASZ PUNEDWY HHAAS зиви PurEVO-15 PurNH=15иTO/50 PORES3 INN RECT POLESSIO BUZH FORTALTOSHO NEXT HOLD FIREING ST gir illi tilt. I 4000 AB=INT(PRD(1/#10) 4016 HU-INTORNICI (#416 (#44 4020 IFHB=SUPHB=6THFNPUNEOW+HU,56 PD FEODE JS +HILL 4 4850 BURRUHI PRINT"[BUMICR/ OBJECTOR). SCORE 1 OF 1EPPOTIBLE 14 PULL TUBE 16 4055 IFAUV=1STHENHURU 4060 POKEOWHAR, SY POKESSHOWHAR, 4 RET 1 feta 4560 PUNES3#1 11 4565 POLEOWARH 57 4510 FORFI-15 FOUSIEP+1 45.00 FOR ELOURE 45 at FORT-ITURBURNET 4º 40 le aler 4556 FOLC of H. 4566 Helleuist 45 0 PUKEUW+X+Y*02,58 =1 Y=19 T14-1 हु१(1) म्हासून स्टब्स 4575 PUREUM+AA,58 4580 RETURN 4995 CR=CR+1 TY=TY+2 10=10+2//00 RE1U 5000 POKE36869,240 PRINT"[CLR]""PUKE V0+1,110 5010 FURKE=1T010 FURT=1T0250 NEXT 5020 PRINT"[CLR]" FORT=1TO210 NEXT P PINT*EWHTIC. ROJASESPCIPHESENT [HUMIE18CPSPD] [HORSEK]SOPPEN "JOH NE 'T SHAS PRINT"[CUR][100RSRD][RVSON][YEL] ####EHBYPINTHESPEJPHUK####EBLSJERVSCFFJ P. FORT-ITICORD NO. SING TIB="BRANCHER! RETURN 6000 PCFE3656) 240 PG() 647-105 6010 PRINT"[CLR][20RSRU][RVSON][YFL] ####LHEVRINTH(.F.)PHLF####LBJUJ[RVSGEE] FOLD PRINT"[CR: RD][WHT]SCORE " S FOR METERSON REPORTED IN THE LIKE CASE SOAU PRINT"[CRSKD][MHT]REFORD ".PEC SOSU PRINT"[CRSKD][MHT]FEDIT ".CR 6060 PRINT"CURSPOONMHT3MNOTHER(SPC)G HME ESPECIES/NUESPECIP" EN70 DETM\$ IFA\$=""THENEU70 6H80 IFH\$="S"THEH450 60-01 END 6500 FORJJ=1TOZ5 POKEYO,15 POKES3..4 6510 FORT=1TO25 NEXT 65aU POKESS U. 6530 FOR 1=1 TO 5 NE F REAT 6540 CHALLERSON CIACIAL RETURN 7000 PUKEVO, 15 7018 FUREL=1T0100 POKES3, INT(RND(1)* 178 (+1.) JUQU FORT-ITOID NEXT NEXT ZHEM PEKEYO, M. POKESS, O. 7040 St=SE+500 RETURN

C-64

Astral



Ignacio Ortega Ortigosa Virgen del Carmen, 19 d/2º B Algeciras - Cádiz Estas lineas van dirigidas más especialmente a los que, como yo, sois novatos en el arte de ordenar ordenadores. Los otros,

los de mayor y mejor nivel, también pueden leerlas, pero que sean indulgentes con lo que para ellos pueden ser herejías informáticas.

A la vista de las cifras de venta de máquinas CBM, pienso que no soy el único novel. A mí me pusieron los Reyes un 64 y entré en faena. También me suscribí al encantador Mundo del Commodore y empezaron las sorpresas

Algunas personas me decian que un ordenador es una máquina tonta a la que hay que enseñarle todo. Primera sorpresa: La máquina no es tonta; la máquina es inflexible Y me liamaba tonto a mí, con la delicada frase: Syntax error... ¿A vosotros también?

Pero no hay que desanimarse. Adelante. Comprendiendo o más bien intuyendo que el Manual del Usuario es un librito escueto con el que se consigue poca cosa más que cambiar grados centigrados a fahrenheit y unas insulsas pelotas rebotando, pasamos a adquirir la Guía de Referencia, joh maravilla! 3.000 pelas. Ahí empezamos a entrever que hay otro mundo detrás de otras puertas.

Los autodidactas continuamos insistiendo. E intentado aprender. En éstas, el cartero llama y nos trae el primer número de Commodore World. Como fieras transcribimos a nuestro teclado los progamas. ¡Caramba!, ¿pero se pueden hacer cosas como el Editor de Sprites de José Ramón Lasa? ¿Cuándo llegaré yo a eso? Leemos el Editorial, en profundidad y nos llega al alma sobre todo el párrafo de crear un Club de Amigos de todas las edades y de admitir cualquier idea o programa más simple.

¡Âh! Esto es muy bueno. ¿Debería yo enviar ese programilla que soluciona los problemas de velocidad, tiempo y espacio? A lo mejor ya está muy visto... Reflexionamos que es posible que nuestro mejor logro, nuestro mayor éxito en programación de novatos está ya más que hecho y superado. Y continuamos el estudio de la Guía de Referencia, con la pregunta en pie. ¿Envío o no envío el programilia?

La Guía, entretanto, se encarga de amargarnos. Ya está habiando de cosas como Selección de Banco y Adaptador complejo de interface, ¡Qué barbaridad! Bueno, nosotros a lo nuestro. Vamos a copiar el ejemplo de los dibujos en alta resolución a ver qué pasa. ¡Sopia!... Ya me he cargado el ordenador. Pero no. Aquí dice que la pantalla se llena de porquería. Entonces debe ser normal. Menos mal que el libro ha avisado. Sigamos. Ahora la curva del seno. Pues no funciona. Tardamos algún tiempo en comprender que los puntos de la sinusoide deben esperar su turno para salir. El libro debería haberlo advertido para no darnos otro susto. Ahora entendemos por qué nos decían que el Basic es lento, cuando observábamos que era capaz de imprimir la tabla de multiplicar en un periquete.

En éstas llega el número 2 de la Revista. ¡Qué voy a enviar yo mi miserable programilla! Menudo listado tiene Manuel Torralba a sus trece años con el Topo Loco manejando pokes y peeks a todo plan. Y, entretanto, la Guía de Referencia haciendo de las suyas. En la página 88, después de hacernos teclear ocho sentencias, nos dice textualmente: ¡Usted acaba de crear un signo @ en color invertido!... Pues no veo yo...

En fin, otra vez en alta resolución y después de averiguar que al libro le falta un next, pintamos un semicirculo en la pantalla... por fin.

No nos desanimemos amigos, sigamos insistiendo en el estudio y machacando teclado, que siempre algún fruto sacaremos. Y como ayuda-guía os envío este programilla que bueno o malo, funciona, dedicado especialmente a aquellos que sin grandes conocimientos ni experiencia como yo, se han desilusionado en principio al ver que el ordenador no perdona cuando te equivocas.

Me anima enviarlo también el ver que poco se ha tocado hasta ahora en las posibilidades gráficas del 64 y que con los dos ejemplos del libro tampoco hay para mucho.

Siento no poder prestar un arreglito para los Vic-ciosos del 20, pero mis modestos conocimientos me incapacitan para eso.

Novatos... animaros a enviar cosas. Aunque os parezean superadas por hace ya tiempo, nos pueden servir a los demas principiantes.

Y ahora al trabajo:

Rutina 1.—Lineas 1000 a 1040. Todo esto está calcado de la Guia de Referencia. Es lo que nos asustó cuando lo pusmos en marcha la primera vez. Coge la pantalla y la hace un churro pero eso si con un divertido coloreado. Luego se formaliza y es como si quisiera dos telones. Podéis cambiar en la linea 1030 en el pokel que está en 3, el código de vuestro color favorito (negro no vale).

Rutina 2.—Líneas 1100 a 1115. También fusitadas de la Guía. Es la fórmula que da el libro para pintar un punto determinado en la pantalla y encima diciéndote que el Basic puede invadirlas.

Rutina 3.—Lineas 2000 a 2350. Es el llo que me he inventado para pintar líneas a voluntad punto a punto. Seguramente se podrá hacer con menos instrucciones o por otro procedimiento más fácil. No lo sé y me da igual, porque éste funciona y es mejor



Tele Sant Just
Mayor, 2 Tel (93) 371 70 43
SAN JUST DESVERN (Barcelona)

INTERFACE para recibir y transmitir CW y RTTY en el VIC-20 y COMMODORE 64

INTERFACE para poder conectar cualquier cassete a los ordenadores COMMODORE 64

BASE DE DATOS en cassette ARCHIVO DE OSI

"NECESITAMOS DISTRIBUIDORES"

que el del libro que deja puntos separados en algunas zonas Este pinta rectas y, con un poco de astucia, unas curvas preciosas

Programa para utilizar estas rutinas

Línea 20.—Sitúa el centro del dibujo en el centro de la pantalla. Si mueves esos valores el dibujo sale en el punto elegido Líneas 70 y 80.—Se hace el reparto de vértices simétricos y

determinan la dirección del segmento

Y todas las demás hacen el cálculo de los puntos y direcciones que han de llevar de una forma continua

Podéis volver a la pantalla normal con Stop y Run/stop-

restore

Si queréis el ambiente más cargado podéis añadir 225 NP = NP + 1 : GOTO80

La coordenada vertical tiene 200 puntos, por lo que os aconsejo que el radio exterior no supere el número 100 porque si no el dibujo queda incompleto. Pero si lo hacéis, tampoco se rompe nada (veréis que esto de las averías me preocupa mucho y es que los Reyes están todavía pagando las letras).

Me gustaria que Pere Masats, artista neurocirujano informatico, nos fabricase unas rutinas en CM para que todo este lio de cosas fuese menos lento, especialmente las rutinas 1 y 2 que son

los desesperantes cambios de decorado.

Fengo otras cositas monas sobre el mismo tema. ¿Las quereis? No dejéis de trabajar, insisto, y mandad lo que tengáis por tonto que sea. Yo he aprendido más de las cosas de otros, sobre todo las facilonas, que del libro

```
1 REM......ASTRHL......
6
8 '
10 PRINT"CCLR3"
20 CX=160 CY=100
30 INPUT"[3CKSRD][2CRSRR]NUMERU[SPC]
DETSPOIPTOOS", NE
40 INPUT"[2CRSRD][2CRSRR]RADIO[SPC]E
KTERIUR" RM
50 INFUT"[ZCRSRD][ZCRSRR]RHD[GESPC][
HTERIOR", RN
PO GOZDRIGO
78 RN=2411/NP
BU DH= IT/NP
98 X1=CX+RM#COS(AN)-Y1=CY+RM#SIN(AN)
100 FORI=1TONP
110 AN=AN+DA
120 X=CX+RN#COS(AN) - Y=CY+RN#5IN(AN)
130 DX=X-X1 DY=Y-Y1
```

```
160 XI=X YI=Y AN=AN+DA
170 X=C/+RM#COS(AN) Y#CY+RM#SIN(AN)
180 LAEX-X1 DYFY-Y1
190 X=X1 Y=Y1
200 G05UB2000
218 Klax Ylay
ZZW NEXTI
2.0 G HU230
999 ENL
1000 REM
             PASE A BIT MAPPED
           "[20C0MMY]"
1001 REM
1005 BASE*204096 POKE53272, PEEK (5927
2) OR8
1010 POKE53265, PEEK (53265) OK32
1020 FORT=BHSETOBASE+7999 PUKET.0 NE
1030 FURT=1024T02023 POKET, 3 NEXT
1040 RETURN
1042
1098 REM....CHLCULO DEL PUNTO....
1099 REM "C21COMMY1"
1100 CH=1NT(X/8)
1110 RU=INT(Y/8)
1120 LN=YAND?
1130 BY=BHSE+RO#320+8#UH+LN
1140 B1=7-(XAND7)
1150 POKEBY, PEEK (BY) OR (2181)
1155 RETURN
              CALCULO DEL SEGMENTO
1998 REM
1999 REM"[25PC][23COMMY]"
2000 X1=X-Y1=Y
2010 X2=X1+DX Y2=Y1+DY
2015 IFABS(DX))HBS(DY)THENZZOU
2020 IFDYKOTHEN2100
2030 FURS-Y1T0Y2
.040 GOTO2110
2080 RETURN
2100 FORS=Y1TUY2STEP-1
2110 X=(X1+DX/(-BY)#(Y1-S))
2120 Y=S
2130 GUSUB1100
2140 NEXTS
2150 RETURN
2200 IFDXCOTHEN2300
2230 FORS=X1TOX2
2240 GOTO2310
2260 RETURN
2300 FORS=X1TUX2STEP+1
2318 Y=(Y1+DY/(-DX)#(X1-S))
2320 X=5
2330 GOSUBII00
2340 NEXTS
2350 RETURN
```



140 X=X1 Y=Y1

150 GOSUB2000

Asterisco

VIC 20 SIN AMPLIACION

Higinio Porta López - 16 años Avenida Huesca, 20 Sariñena (Huesca) Como es el primer programa un poco largo que hago, tal vez lo encontréis un poco hado y desordenado; pero como

funciona, ya vale.

El programa ocupa casi toda la memoria del VIC, deja 36 bites libres. Es una pena que el VIC tenga tan poca memoria Por esa falta de memoria no he podido hacer la presentación meior.

Ahora paso a explicaros de qué va:

Es un juego parecido a los de marcianitos, pero no os asustéis, cambia en algunas cosas. (Espero que no hayáis aborrecido ya a

los pobres marcianitos.)

Manejas la nave por toda la pantalla persiguiendo al marcianito—sólo hay uno—. Debes disparar y darle, es decir, hacer blanco en el.

Al principio dispones de 3 tankes (naves) y 8 tiros —que puedes aumentar— por cada tanke. Cada vez que le das consigues 10 puntos y un tiro más; si fallas, lógicamente, pierdes el tiro. Cuando agotas los tiros pierdes el tanke, y si agotas los tankes...

Todo esto se explica en la presentación del programa, pero aún hay más: A los 200, 400 y 800 puntos consigues 2 tankes extras. ¡Ah!, y si consigues el récord, pones tu nombre.

Nada más, sólo me queda animar a todos para que manden colaboraciones. Yo creo que ayudan mucho a los principiantes

38/Commodore World Octubre 1984

```
A4#="[SPC]NHDIE"
  POKE36879, 25
5 POKE36878, 15
6 G0SUB11886
10 G=7800 L=8130 A3=0
20 TN-3 TR=8
30 HR=113-AB=114 IZ=115 DE=107
40 PT=0
45 PRINT"[CLR]"
50 FORP=7724T07745, GUSUB100 NEXT
60 FORP=7,746TU8164STEP22 G03UB100 NE
70 FURP=8165T08185 G03UB100 NEXT
80 FORP=8163T077675TEP-22 G05UB100 N
EXT
90 6070110
100 POKEP, 102 POKEP+30720,4 WETURN
110 PRINT"LHOMOCUEDHN", TN, "THNKS", TR
 "ESPOITIR"
120 PRINT"[HUM][CRSRD]PTS",PT,"[35PC]
RECESPOIPTS", HP
130 POKEG, 42 POKEG+30720, 0
140 POKEL, HR PUKEL+30720.0
1000 POKE650, 255
1010 GETHS IFHS "THENZOUG
1626 1FH$="L"THENC=1 POKEL:32 L=L-22
 GUSUB7888
1030 IFH#=","THENE=2 PUREL,32.L=L+22
 GUSUBZUNU
1040 IFAs="Z"THENC=3 POFEL,32-L=L-1:
ดับร<sub>ั</sub>บสินใชยชา
1050 IFH#="X"THENC#4 POKEL:32-L#L+1
ษยรปิติสัยย์นั้
1860 IFH*="[SPC]"THENGUSUB/010 GOSUB
NNNE
1070 IFC=1THENIFPEEK(L)=102THENL=L+2
2 G05UB7020
1080 IFC=21HENIFPEEK(L)=102THENL=L-2
2 60-UB7628
1090 IFC=3THENIFPEEK(L)=102THENL=L+1
 DESUBRULD
1100 IFC=4THENIFPEEK(L)=102THENL=L-1
 GUSUB7020
1110 IFPEEK(L)=42THENPOKEL,42 GOSUB?
040 - GOSUB4200 - GOSUB4010 'L=8130 - POLEL
ARR
1120 IFC=1THENPOKEL, AR POKEL+30720, 9
1130 IFC=2THENPOKEL, AB POKEL+30720.0
1140 IFC=3THENPOKEL, IZ POKEL+30720.0
1150 IFC=4THENPOKEL, DE POKEL+30720, 0
2000 POKEG, 32
Z810 F=INT(RND(1)#5+1)
2020 IFF=1THENG=G-22-IFPEEk(6)=102TH
ENG≃6+22
2040 IFF=2THENG=G+22 IFPEEk(G)=102TH
ENG=0-22
2060 IFF=3THENG=0-1 · IFPEEK (G)=102THE
NG = G + 1
2080 IFF=4THENG=G+1:IFPEEK(G)=102THE
Nú≃G+3
2100 B=PEEK(G)
2110 IFB=10/ORB=1130RB=1140RB=115THE
NFOKE6,42 GUSUB7040 GOSUB4200 L=8130
GUSUB4010 POKEL, HR
2120 POKEG, 42 POKEG+30720, 0 POKE3687
4,246 FORU=1T020 NEXT-POKE36874,0-GO
T01000
3000 N=L Y=0
3010 k=PEFF(L)
3010 IFK=H-1HENM=-22
SULU IFK#HBTHENN=22
3040 IFF ≤I. THENM=-1
3950 IFF=DETHENM=1
BUREL SEFEEN (NOM)
3070 IFS=1021HENGUSUB7050 POKEN-32 €
```

USUB4000 RETURN 3080 IFS=42THENPUKEN,32 GOSUB6000 TR ■TR+1 GOSU87030 GOSUB4010 RETURN 3090 Y=Y+1 3100 IFY<>1THENPOKEN, 32 3110 IFK=ARORK=ABTHENPOKEN+M,93 POKE N+M+ 30720, 0 3120 IFF = 120kk = DETHENPOKEN+M, 64 POKE N+M+30720.0 3130 N=N+M 3140 GOT03060 4000 TR=TR-1 IFTR-0THENTH=TN-1 GUSUB 7040 TP=8 PUREL,32"L=8130 PUREL,AR G 05UB41W6 4610 PRINT"CHOMICIAL RURED" TRUTCH RERET [SPC]" RETURN 4100 IFTN≃0THENPOKE36874,0 60TU5000 4110 PRINT"[HOM3[6CRSRR]", TN, "LURSRL] [SPC]" PRINT"[HOM1[JUKSRETT" KET HN 4200 TH=TN-1-TR=8 IFTN=0THENPUKE3687 5,0 GOTOSMUB 4210 PRINT"[HOM][6CR5RK]", TN, "ELKSRL] LSPC1" - PRINT"[HUM][9CRSRR]T" RETURN SUBU PRINT"[CRSRU][7Ckskk]@[7CRSRR]@ 15PC1" 5001 PRINT"[9CRSRD][6CRSRR]GHME[9PC] OVER" FORU-1704000 NEXT GOSUB10000 5005 PRINT"LULRITURSRUITURSRUITURSHO CSPCIMAS?" 'PRINT" [HOM3[30RSRD1[CRSRR] (\$/N)" PONES6878,0 5010 DET2\$ 1F24=""THEN5010 Suzu IFZ#="S"THERZ 5030 IFZ# - "N"THENEND 5040 60T05010 5999 REMPUNTUS###### 6000 PT=P1+10 IFPT>HPTHENGOSUB9000 R P=PT GU: 86100 6005 IFPT=L00THENGOSUB8000 PONE36879 .24 TN=TN+2 GOSUB4210 6007 1FPT=400THENGOSUB8000 POKED6879 ,31 TN=TN+2-60SUB4210 6008 IFPT≃800THENDUSUB8000 PUNE36879 .88 TN=TN+2 5U5U£4210 6010 PRINT"[HOM][CRSFD][3CRSRR]",PT RETURN 6100 PRINT"[HOM][CRSRD][17CRSRR]",HP ME TURN 7000 POKE3US75-2:0 FORU=1T010 NEXT P OFE 6875, @ FETURN 7010 POKE36877,237 FORU=11050 NEXT P OKE36877,0 RETURN 7020 FORE36976,150 FORU=1T020 NEXT P OKE36876,0 PETURN 7030 FORU=15T005TEP-1 PUKE36876,232+ U FORR=1T010 NEXTR/U POKE36876,0 RET LIKH 7848 FORUS-301055 PURE36876, 288+U NE XTU PURE 16876. W RETURN 7050 POKE36874,242:FURU=1T050 NEXT R ETURN 8000 PRINT"[HOM][13CRSRD][8CRSRRJBUN 3010 FORA1=1T030 POKE36876, INT(RND(1)#128)#128 FORAZ=1TO10 NEXTAZ/A1 PUK E36876.0 SUZU PRINT"[HOM][13CRSRD][BCFSRR][55PC] " RETURN 9000 A3=A3+1 IFA3=1THEN1FHPC>0THENPR INT"LHUMILISCRERDI[5URSRRINUEVO[5PU] RECORD" GOTO9010 9005 RETURN 9010 FORH1=1T050 POKE36876, INT/RND/1 >*128)+128-F0RAZ=1T010 NEXTA2/A1*P0k E36876.0 9020 PRINT"[HUM][13CRSRD][5CRSRR][12SPC] RETURN 10000 IFRPCHPTHENRP=HP-60T010020 10010 IFRPKHPTHENRP=HP:GOTO10020 10015 GUT010100

10020 POKE36878.0 PRINT"[CLR][CRSRD] [GRSPRITUISPOITIENES[SPCIEL[SPCIRECO RDI" POKE36879,156 18830 PRINT"LHOMICS F RDICLRSKRIESCR INEESPORTUESPORNOMERE" 10040 INPUT"EHOMILSCREPDIECE SERI", 84 16050 PRINT"[CLR][URSRD][60RSRR]RECO REPORT 10060 PRINT"[HOM][9CRSRD][7CKSRR]",A 10070 FORU=1T05000 NEXT 10080 RETURN 10100 PRINT"CCLRICORSRDICORSPRILOCSPOI SIENTO, ESPERNOESPECITIENES" PRINT PPI NT"EBSPECIRECORD" PORE 36-73 154 10110 PRINT PRINT"[HOM][5CRSRD][7CRSRR] LOISPOITIENE [CRSFD]" 10120 PRINT"[HOM][7CRSRD][7CRSRR]", A 4\$ FORU=1T05000 NEXT 10130 RETURN 11000 PRINT"[ELR][GRSPD][3CRSkk]HHZ[SPC] BURHLUISPU TEHLSPU JEL

11010 PRINT"[CPSRD][6CPSRR]HSTERISCU 11020 PRINT"(CREPDICORERRIZ=12DA." 11030 PRINT"[55PC]X=DELHA." 11040 PRINT"[CRSRD][LRSRR]L=AFRIBH[45PC] PHEH TU! 11050 PRINT"[CRSRD][3CRSRR]E=PACIO=D I SEHRO! 11060 PRINT"[CKSRU][CRSRR][IENES[SPC] BESPORTHNI EST 25PC 1Y" Times PRINT TOW-RELEASERIST FOITIR Out follenkt fouthful 110KO FRINT [CRUMD][CAUAR]CADHCUPC]B LHNCO ' 11090 PRINT"[CRSPD][7CRSPR]][SPC]TIR OLSPE IMHS" 11100 PRINT"[CRSRD][6CKSRR][UESPC]PU NTUSCSPU IMHS 11110 PRINT"[CRSPD][3CRSRR]PULSHLSPC] UNHESPECTECLA" 12000 GETHES IFHES=""THEN12000 12001 RETURN



Prints sofisticados

00777444660055777446666477764401144868800740754477747774441150005770442160005777446681120044774441114774444154

José Luis Muñoz Franco Monterde, 1-7º B Zaragoza - Tel. 34 00 70 Os remito un programa, en realidad tres subrutinas, que creo que servirán como complemento al artículo publicado en

el número 3 de abril titulado PRINTS SOFISTICADOS. Están hechas para su uso en un Commodore 64, aunque creo que son fácilmente adaptables al Vic. Lo primero que quiero puntualizar es que no he querido agrupar líneas de programa mediante la técnica de los dos puntos para hacer más fácil su comprensión, por lo que cada uno es libre de agrupar las líneas que pueda. Así mismo he separado lo que son palabras Basic y variables dejando un espacio entre ambas, también por el motivo anterior. Y paso ya a explicar un poco el programa

Comienza en la línea 10 solicitando Print 1, 2 ó 3.

Linea 20: Filtra los caracteres que no sean 1, 2 ó 3 no dejándolos pasar.

1 mea 25 Pide dar entrada a una cadena.

Linea 30 Desvia la ejecución del programa hacia una de las subrutinas elegidas.

Linea 40: Bucle de tiempo antes de volver a ejecutar de nuevo el programa.

El programa no tiene más. Ahora paso a explicar las subrutinas.

NUMERO 1: IMPRIME ALTERNANDO UN CARACTER DE IZQUIERDA Y OTRO DE LA DERECHA DE LA CADENA.

Linea 1010: Mide la longitud de la cadena.

Linea 1020: Si la longitud es impar añade un espacio por la izquierda a la cadena. Si es par añade un espacio por la izquierda y otro por la derecha.

Línea 1030: Înicia un bucle para imprimir los caracteres de la cadena.

Linea 1040 a 1070: Se encargan de imprimir un carácter de la izquierda y desplazan el cursor los lugares necesarios a la derecha.

Linea 1080 a 1090: Imprimen carácter de la derecha Linea 1095: Vuelve para imprimir otros dos caracteres.

Si os habéis fijado, la subrutina no trabaja imprimiendo los caracteres de uno en uno sino que primero imprime uno de la izquierda y uno de la derecha, luego 2 de la izquierda y dos de la derecha, etc...

NUMERO 2: DESPLAZA CADA CARACTER DE LA CADENA A LO LARGO DE UNA LINEA Y LUEGO REPITE POR LA LINEA SIGUIENTE HASTA QUE EL CARACTER OCUPA SU LUGAR

Línea 2010: Inicia el bucle para la impresión de todos los caracteres

Linea 2020: Si el carácter es un espacio establece V en cero para acelerar el bucle de tiempo de la linea 3060 y 2060.

Linea 2040 a 2080. Desplazan el carácter correspondiente por la linea superior

Línea 3000 a 3080: Desplazan el carácter por la línea inferior después de haber bajado una línea el cursor en 2090

Linea 3090: Imprimen el carácter definitivamente.

Linea 4000: Sitúa el cursor para en la línea 4010 volver a tomar otro carácter

Creo que se puede adaptar para el VIC si en la linea 2040 ponemos 21 y no 39. En la 3000 ponemos lo mismo.

NUMERO 3: IMPRIME UNA CADENA. LUEGO TOMA CADA CARACTER Y LO VA DESPLAZANDO POR LA LINEA DE ARRIBA HASTA QUE LO COLOCA DELANTE DE LA CADENA QUE HA IMPRIMIDO.

Línea 5010: Mide la cadena y asigna a C un valor que es el del doble de la longitud de cadena.

Linea 5020: Imprime la cadena en el lugar correspondiente. Linea 5030: Establece bucle para impresión de caracteres.

Linea 5040: Sitúa cursor sobre carácter a desplazar, lo borra y lo sitúa en la linea superior.

Linea 5050: Establece bucle para desplazar el carácter.

Linea 5060: Imprime carácter

Linea 5070: Bucle de tiempo.

Linea 5080: Borra arácter del lugar en que está, para en la 5090, seguir con el bucle de desplazamiento.

Línea 6000: Sitúa definitivamente el carácter en el lugar que le corresponde. A la variable C que se utilizará en la línea 5040 para situar el cursor sobre el siguiente carácter a desplazar, le resta 1.

Linea 6010: Vueive para tomar otro carácter.

Y nada más, me despido esperando que os hayan gustado estas tres subrutinas.

```
6 REM
       *** SUBRUTINAS PRINT ***
       *** POR J.LUIS MUNG2 ***
 REM
       淮南水南南南南南南南南南南南南南南南南南南南南南南
B REM
10 PRINT"[CLR]" PRINT "[9SPC]SOLICIT
AESPCIPRINTESPC11-2-3"
20 GET C$ IFC$= "1" OR C$= "2" OR C$
- "3" THEN25
22 GOTO 20
25 INPUT"[2CRSRD][9SPC]UNR[SPC]PALAB
RAISPOI 'ANS
27 PRINT"(2CR6RD)"
30 ON VAL(C#)GOSUB 1000,2000,5000
40 FOR L=1 TO 600 NEXT L GOTO 10
1000 REM SUBRUTINA PRINT NUMERO 1###
****
1010 L=LEN(A$>
1020 IFL/2 () INT(L/2) THEN AS="[SPC] "+AS:L=L+1 GOTO 1030
1025 As="[SPC]"+As+"[SPC]".L=L+2
1030 FOR QU=1 TO L/2
1040 PRINT LEFT*(A$,QQ) PRINT"[CRSRU]
1050 FOR KK=1 TO 90 NEXT KK
1060 FOR MM=1 TO L-1.PRINT"[CRSRR]";
NEXT MM
1070 L=L-1
1080 PRINT RIGHT $ (A$, QQ) : PRINT "[CRSRU]
1090 FOR KK=1 TO 90 NEXT KK
1995 NEXT QQ
1998 RETURN
2000 REM SUBRUTINA PRINT NUMERO 2***
***************
法律未未申申法
```

```
2010 FUR MM=1 TO LEN(A$)
2020 IF MID#(A#.MM,1)="[SPC]" THEN V
=0 GOTO
             2040
2030 V= 50
2040 FOR QQ= 1 TO 39
2050 PRINT MIDS(AS,MM,1);
2060 FOR KK=1TO V-NEXT KK
2070 PRINT"[CRSRL][SPC]";
2080 NEXT Q0
2090 PRINT"[CRSRD]",
3000 FOR QQ=1 TO 39-MM
3050 PRINT MID#(A*, MM, 1);
3060 FOR KK=1 TO V NEXT KK
3070 PRINT"[CRSRL][SPC][2CRSRL]";
3080 NEXT QU
3090 PRINT MID$(A$,MM,1)
4000 PRINT"[3CRSRU]";
4010 NEXT MM
4020 RETURN
5000 REM SUBRUTINA PRINT NUMERO 3###
*************
中华中华南北市
5010 L= LEN(H$)-C= L#2
5020 PRINT TAB(L+2); A$
5030 FOR MM= L TO 1 STEP -1
5040 PRINT"[CRaRU]", TAB(C+1), "[SPC][CRSRU]
[CRSRL]
5050 FOR QQ = L TO 0 STEP -1
5060 PRINT MID$(A$, MM.1)
5070 FOR KK=1 TO 70: NEXT KK
5090 PRINT"[CRSRL1[SPC][2CRSRL]",
5090 NEXT QQ
6000 PRINT"[CRSRD]"; MID#(R$,MM,1):C
=C-1
6010 NEXT MM
6020 RETURN
```



Aplástalo y portadas

Román Estébanez Santesteban 16 años - S. Fermin, 49 - 1º izqda. PAMPLONA (Navarra) Tel. 236 372

Me Ilamo Román Estébanez y vivo en Pamplona. Poseo un commodere 64, un cassette y una impresora MPS-801 Os man-

do estos trucos y dos programas que espero sean de vuestro agrado.

Los trucos son los siguientes:

I. Este truco facilita el paso en la impresora del modo mayúsculas y gráficos al de minúsculas. Consiste en pulsar dentro del PRINT# la tecla del cursor hacia abajo. En la pantalla aparecerá un Q en video inverso. Esto hará que la impresora comience a escribir en minúsculas sin necesidad de usar ningún

Para volver al modo mayúsculas basta con pulsar dentro de las comillas la tecla del cursor hacia arriba

Se trata de un truco para facilitar la búsqueda de un programa en el cassette. Basta con colocar un cable desde la patilla Nº 5 de la salida audio/video al cable Nº 4-d (blancolectura del cassette) en el port del cassette.

Abriendo el volumen con un POKE 54296, I podremos monitorear a través del altavoz de nuestro televisor lo que hay grabado en la cinta, ayudándonos a encontrar los programas por su pitido micial.

NOTA: Este truco es muy sencillo en el cassette antiguo, ya que basta con introducir un cable pelado por el hueco que deja el cable blanco en la parte trasera del conector. En el modelo moderno habria que desmontar la clavija y hacer la conexión oportuna

Aplástalo

El juego consiste en aplastar a todos los "malos" posibles en el menor tiempo. Tú conduces al último ejemplar de la raza humana, y en un desesperado intento, has de aniquilar a todos los enemigos que se crucen en tu camino. Dispones de dos minutos

El procedimiento para aplastarlos es empujando unos enormes bloques de piedra de manera que quede aplastado por el enorme peso y perezca..., pero eso no es todo. Tu enemigo posee varias vidas y se reencarnará en otro punto de la pantalla hasta que se agoten sus "nuevas vidas". Por cada pantalla que logres superar, el número de reencarnaciones se incrementará, haciendo más dificil tu labor (este número se indica en la parte superior derecha de la pantalla)

Al término de cada pantalla se mostrará el tiempo invertido y una bonificación acorde con el tiempo

Otro procedimiento de obtener puntos es juntando los tres valiosos diamantes dispersados por la pantalla. En un principio son de color amarillo, para que puedas divisarlos fácilmente, pero cuando los mueves retornan a su color negro. Si lo consigues lograrás 10 000 puntos

En principio dispones de tres vidas, indicadas en la parte superior central de la pantalla. Si bates la puntuación récord, el ordenador te pedirá que entres tu nombre para figurar así en la tabla mundial de aplastadores. ¡Animo y suerte!

DESCRIPCION DEL PROGRAMA

Lineas

Comentario

10 - 160Inicialización y generación de caracteres. 200-290 Inicialización pantalla de juego.



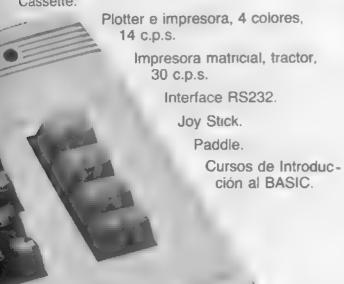
Cuando se tiene 64 K de memoria, una magnífica resolución, 16 colores, efectos tridimensionales con sprites, un sonido equivalente al de un sintetizador, un teclado profesional con 62 caracteres gráficos, toda una amplia gama de periféricos, la más completa gama de programas educativos, profesionales y de video-

juegos...; en resumen, cuando se es un ordenador personal como no existe ningún otro en el mercado y el más vendido mundialmente, es muy difícil decir sin orgullo que eres un Commodore-64.

Claro que más difícil todavía es decir sin orgullo que tienes un Commodore-64. ¿Por qué no lo comprueba?

COMMODORE 64 LE DA ACCESO A MUCHOS ACCESORIOS

Unidad simple de disco (Monofloppy) 170 K. Cassette.



COMMODORE 64 LE MUESTRA PARTE DE

Utilitarios y lenguajes

MONITOR LENGUAJE MAQUINA.

FORTH.

MACRO ASSEMBLER. PROGRAMMER'S

UTILITIES.

LOGO. TURTLE GRAPHICS II.

PILOT. MASTER.

Sistemas operativos

FILE/BOSS.

CP/M.

Programas de aplicaciones

EASY SCRIPT.

Proceso de texto de gran potencia.

CALC RESULT.

Hoja electrónica de cálculo.

EASY CALC RESULT.

Versión simplificada del CALC RESULT.

MAGIC DESK

Proceso de texto y gestión de ficheros.

AGENDA TELEFONICA.

Programas educativos

MUSIC MACHINE.

MUSIC COMPOSER.

GEOGRAFIA I. GEOGRAFIA II.

VISIBLE SOLAR

JUEGOS EDUCATIVOS.

SYSTEM.

TEMAS

SPEED/BINGO MATH FISICA I.

MONOGRAFICOS. CONOCIMIENTOS

MATEMATICAS I.

GENERALES.

HISTORIA I.

QUIMICA I.

Juegos

JUPITER LANDER.

FROGMASTER. GRID RUNNER.

KICKMAN.

SEAWOLF. RADAR RAT RACE.

ATTACK

OF THE MUTANT

TOOTH INVADERS.

CAMELS.

LAZARIAN.

THE PIT. MR. TNT.

OMEGA RACE.

6 GAME PROGRAMS.

LE MANS. PINBALL

BINGO.

SPECTACULAR.

ROOTING TOOTING.

AVENGER.

MINESSOTA FAT'S

SUPERMASH.

POOL CHALLENGE.

... v seguimos ampliando la lista

El ordenador personal de la familia más potente

z commodore

MICROELECTRONICA Y CONTROL, S.A. c/. Taquigrafo Serra, 7, 5.º. Barcelona-29 c./ Princesa, 47, 3.º G. Madrid-8

300-540 540-670 700-730 1000-1150 2000-2030 2500 3000-3080 3500-3520 4000-4040 4100-4130 4300-4320	Rutina de juego. Controlador del movimiento del bloque de piedra. Examina posición de los diamantes. Presentación del juego. Bonos extras por juntar los diamantes. Actualiza la puntuación. Pantalla de tiempo realizado en el juego Actualiza marcador de supervivencia (Nº de vidas) Rutina variable de sonido Sonido de muerte del humano Reencarnación del "malo".

5500-5510 6000	Fin de tiempo. Rutina de inicialización del sonido
6500	Cálculo de números aleatorios
7000-7030	Localización de la reencarnación (posición en pantalla).
8000-8020	Pantalla de récord batido.
9500-9520	Datas del humano, malo y bloques.

Este programa usa el joystick en el pert. 2. Moviéndolo en las 4 direcciones te desplazas por la pantalla, y apretando simultáneamente el botón de disparo empujas al bloque en la dirección deseada.

Portadas

Este sencillo programa sirve para hacer portadas para los cassettes, de manera que sean fácilmente reconocibles.

Al ejecutarlo pide por pantalla el nombre del programa que va a ocupar la primera cara, luego el de la segunda y finalmente el nombre del autor o autores

Automáticamente la impresora empieza a realizar la tapa, marcando los limites superiores, inferiores y lateral derecho. El izquierdo queda delimitado por el propio papel. El texto que imprime es susceptible de modificación. Yo opté por esta presentación por gusto particular.

```
APLASTALO
          前班事业统
10 0Q=53280 V=54272:POKEV+24,15.DIM
SP(15):H#="CBM(SPC]64":HI=20000
20 FOR R=1 TO 5: READ CO(R) - NEXT R: FO
RR=1 TO 12 READ SP(R) NEXT R
40 D(1)=1.D(2)=-1.D(3)=40.D(4)=-40 S
S=1024 GC=12288 JO=56320
60 IF PEEK(GC)=56 THEN 160
70 POKE52,48 POKE56,48
80 POKE56334, PEEK (56334) AND 254 PRIN
T'[CLR][4CRSRD][45PC][RVSON]GENERAND
OISPOJCARACTERES'
90 POKE 1, PEEK(1) AND 251
100 FOR I+8 TO 1232 POKEGC+I, PEEK(I+
53248) NEXT
110 FOR T=216 TO 247 READ A POKEGC+T
A NEXT
120 FOR I=GC TO BC+7-READ A-POKEL, A:
NEXT I
130 FOR I=1280 TO 1287 READ A POKEI+
GC, R. NEXT
135 FOR I=648 TO 655 READ A POKEI+GC
A NEXT
140 POKE 1, PEEK(1) OR 4: POKE 56334, PE
EK(56334)0R1
160 POKE53272, (PEEK(53272) AND240)+12
:POKE 53281,0:POKE 53280,6:GOTO 1000
200 PRINT"[CLR][2CRSRD][2SPC][COMMA]
[33SHIFTC][COMMS]"
220 FORT=1TO20 PRINT"[2SPC][BLK][SH1FTB]
CBLK1C33SPC1CBLK1CSHIFTB1" - NEXT T
230 PRINT"[2SPC][BLK][COMMZ][335H1FTC]
CCOMMX1"
231 PRINT"[SPC]HIGHESPC]SCGREESPC]:"
HI"...."H$, PRINT"[HOM][SPC]UP1[9SPC]
VIDAS CESPE IMALOS: "
237 GOSUB 3500 FORT=1TONE GOSUB 6500
```

260 POKE SS+X1+40WY1,169 NEXT: IR=0.F

```
R=256 SR=10 NV=1
270 FOR U=1 TO 3
275 X=INT(RNB(0)#31)+4
280 Y=INT(RND(8)#18)+5 IF PEEK(SS+48
#Y)<>32 THEN 275
285 POKESS+X+40#Y,90 POKESS+X+40#Y+5
4272,7 GOSUB 4000 NEXTU
290 X=13 Y=10 TI$="000000"'MM=64 PB=
100 S2=0
300 P=$$+X+40#Y POKE P.0 Q=$$+IN POK
EQ.81
310 RR=PEEK(Q+D(S)) · IFRR=0 THEN 500
315 IF TI$>"000200" THEN 5500
320 IF RR ()32 THEN S=INT(RND(0)#4)+
1 GOTO 310
350 IN=IN+D(S)-O=PEEK(JO): IF O=127 T
HEN FORR=1T030 NEXTR GOTO 440
360 IF (OAND8)=0 THEN X=X+1:N=1.CA=2
2 GOTO 410
370 IF (OAND4)=0 THEN X#X-1:N#-1:CA#
28 GOTO 410
380 IF (ORND2)=0 THEN Y=Y+1 N=48-CA=
30 GOTO 410
390 IF (ORND1)=0 THEN Y=Y-1:N=-40 CR
=29 GOTO 410
400 GOTO 440
410 IF (PEEK(P+N)=160URPEEK(P+N)=90)A
NII(DAND16) -OTHENXX=N POKEP, CA GOSUB6
420 POKEP, 32 IF SW=0 THEN SW=1 POKEP
0-XX,42:FORR=1T060 NEXT R-POKEPO-XX,
32
430 IF PEEK(P+N)(>32 THEN X=X-N-00TO
 440
440 POKEQ, 32 GOTO 380
500 POKEP, 81 POKE Q, 32 NV=7 GOSUB 41
 80 VI=VI-1 IFVICTHEN1000
 540 GOTO 200
 600 GOSUB 6000 POKE V+4,129
 610 PO=P+XX SW=0 PA=PEEK(PO):II=INT(
 RND(0)44)+5.4U=0 POKEP0+54272,0
 620 PO=PO+XX: YU=YU+II - IFPEEK (PO)=81
 THEN S1=1 MP=81 GOTO 640
 630 IF PEEK(PO) (>32 THEN 8000
 640 POKEV+1, YU POKEPO, PR. POKE PC, CO.
POKE PO-XX, 32 SW=1 GOTO 620
 660 IF PEEK(PO-XX)=90 AND S2=0 THEN
  GOSUB 700
 670 POKE V+4,0 RETURN
 700 W=PO-XX' IF S2=1 THEN RETURN
 710 FOR T=1 TO 12 STEP 2
720 IFPEEK(W+SP(T))=90ANDPEEK(W+SP(T
 +1))=90THENFORR=0T09.GOSUB2000:NEXT.
 RETURN
 730 NEXT T-RETURN
 1000 IF SCHI THEN 9000
 1001 POKE V+24,15+64 POKE V+22,255 G
 OSUB 6000 POKE V+4,129 POKE V+1,100
 NV=1
 1002 POKE V+23,1
 1003 FOR T=255 TO 10 STEP- .5 POKE V
+1.T POKE V+22.T NEXT T
 1025 PRINT"[CLR][CRSRD][BLK][5SPC]RO
 MANTSPOIESTEBHNEZTSPOIPRESENTH...
R=230 FR=10 SR=-20 GOSUB 4000
  1026 PRINT*[3CRSRD][10SPC]###[SPC]AP
  LASTALOISPC: *** GOSUB 4000
  1030 PRINT"[2CRSRD][BLK][6SPC]][SPC]
  [[SPC]@[SPC]£[SPC]][SPC]...[SPC]ESTE
  [SPC]ERES[SPC]TU. " GOSUB 4000
  1040 PRINT"[CRSRD][3SPC][RED][SHIFTQ]
  (SPC)(BLU)(SHIFTQ)(SPC)(COMM6)(SHIFTQ)
  (SPC)[BLK]...[SPC]ESTOS[SPC]SON[SPC]
  TUSESPOJENEMIGOS" GOSUB 4000
  1050 PRINT"[2CRSRD][4SPC]DESPLAZA[SPC]
  LOSESPEIBLOQUESESPEIBEESPEIPIEDRA" 0
  0SUB 4000
  1060 PRINT"[20RSRD][4SPC]PARH[SPC]AP
```

LASTAR(SPC]AESPC]TUS[SPC]CONTRARIOS. ":005UB 4000 1070 PRINT"[4CRSRD][2SPC]##[SPC]PULS ARI25PC11FUEGO1[25PC]PHRAC5PC]COMENZ ARISPC1##" GOSUB 4000 1000 IF PEEK(JO)=111 THEN 1090 1081 FOR T=SS+54486 TO SS+54495 POKE T.CO NEXT T:CO=CO+1.5 IFCO>30 THEN C $\Omega = \Omega$ 1082 GOTO 1080 1090 IN=172.S=2.S1=0.NB=140 BA=3:VI= 3.PP=20 SC=0 1150 GOTO 200 2000 PRINT"[HOM][CRSRD][4SPC]###[SPC] 10.000(55PC]BONUS[SPC]POINTS[SPC]*** * PCKE 53281,2 POKE 53280,2 2015 IR=200 FR=40 SR=-20 NV=30 GUSUB 4000 S2=1 PORE 3281,8 PORE53280,2 2030 PRINT"[HUM][CRSRD][35SPC]" -SC=S C+10000 GOSUB2500 RETURN 2500 PRINT"[HOM][4CRSRR]"SC RETURN 3000 POKE 53200,2 PRINT"[CLR][10SPC] CLASIFICACION" 52=0 3015 Ms=MIDs(TTs,3,2) Rs=RIGHTs(TTs, 2) 3020 PRINT"[5CRSRD][7SPC]TIEMPO " PR INT"[CPSRD][7SPC]"M\$"[SPC]MINUTOS 3040 PRINT"[CRSRD][7SFC]"R#"[SPC]SEG UNDOS. " : TP=0 3060 BP=(120-VAL(R\$))#100 BM=0 IF M\$ ="00"THEN 9P=BP#2 GOTO 3070 3070 PRINT"[4CRSRD][5SPC]TIME[SPC]PO INTS="BP SC=SC+BP BM=8 NV=10 BA=BA+1 GOSUB 4000 3075 IF BADE THEN BAEN 3080 FORR=1T02000 NEXTR POKE 53200.6 GUTO 200 3500 PRINT"[HOM]"TAB(32)"[7SPC]", GO SUB 2500 3510 FOR T=1044 TO 1043+VI POKET, 0'N EXT 3520 FOR T=1055 TO 1055+BA-BM POKET, 81 POKET+54272,0 NEXT RETURN 4000 DOSUB 6000 FOR R-ITONY FOR THIR TOFRSTEPSR 4040 POKEV+1.T.NEXTT.R:POKEV+4.0.RET URN 4100 GOSUB 6000 4120 FOR T=200 TO 0 STEP -16 4130 FOR R#T TO T+10 POKEV+1.R NEXT R. T. POKEY+4.0 RETURN 4300 POKE Q+54272,0 PB=(218M)+100 IR **=5 FR**=60 SR=5 4310 FORU=1 TO 5 POKE Q,CO(U) GOSUB 4000 NEXT U 4320 POKE Q+54272,0 RETURN 5500 PRINT"[HOM][9CKSRD][6CR3RR][SPC] [3CRSRP]##ESPC]FIN[SPC]DEL[SPC]TIEMP DESPECTANT 5510 IR=255 FR=20 SR=-1:00SUB 4000 G 0T0500 6000 POKEV+5,16 POKEV+6,128 POKEV,75 POKEV+4,33 RETURN 6500 X1=INT(RND(0)#33)+3 Y1=INT(RND(0)#20)+3 RETURN 7000 GUSUN 6500 7020 IN#Y1#40+X1.BM=BM+1:Q=SS+IN SC= SC+PB NV=1 7025 IF BM=BR+1 THENTT\$=TI\$ FORT=1TO 130 POKEGQ,T NEXT NB=NB+5 MA=MA+.5 G 010 3008 7030 GOSUB4300 GOSUB3500 GOTO 670 8000 IF S1=1 THEN S1=0 GOSUB 7000 B010 G0TO 660 PRINT"[YEL][COMM1]":NV ■1 'IR=200 FR=20 SR=-10-GOSUB 400 9000 PRINT"[CLR][4CRSRD][5SPC]HAS[SPC] BATIDOISPOJELISPOJRECORDISPOJO P 9010 FORTY=1T02 IR=10 FR=200 SR=10 G

OSUB4000 IR=200 FR=10 SR=-10 GOSUR40 **00 NEXT** 9050 PRINT"[3CRSRD][3SPC]TU[SPC]NOMB REISPOJ"; : INPUT H\$ IF LEN(N\$)>12 THE N PRINT"[5CRSRU]" GOTO 9050 9060 PRINT"[4CRSRD][25PC]TE(SPC]TEND REESPOJENESPOJCUENTAESPOJPARAESPOJLO S" HIRSE 9070 PRINT"[2CRSRD][7SPC]CHMPEONATOS [SPC]DEL[SPC]MUNDO." FORR=1T03000 NE XTR GOTO 1000 9502 DATA 46.43,42,87,81,1,2,1,-1,-1 -2,-40,-80,40,-40,40,80 9505 DATA28,8,15,8,8,24,36,68,56,16, 240, 16, 16, 24, 36, 34, 186, 146, 124, 16, 16 16,40 9510 DATA 108,0,0,0,56,84,146,170,17 0.56,16,124,84,84,16,40,102 9520 DATA 126,195,153,165,165,153,19 5,126,130,68,56,40,126,56,68,130

10 REM PORTADAS DE CASSETTES 20 REM 30 REM ROMAN ESTEBANEZ....22-05-1984 35 REM 50 PRINT"[CLR][2CRSRD][6SPC]PORTADAS CSPCIDECSPCICASSETTE" 60 PRINT"[3CRSPD]NOMBRE[SPC]DEL[SPC] PROGRAMAISPOICHRALSPOIL"; INPUT NI\$ 65 PRINT"[3CRSHD]NOMERE[SPC]DEL[SPC] PROGRAMACSPCICARACSPCI2"; INPUT N2# 70 PRINT"[2CRSPD]NOMBRE[SPC]DEL[SPC] AUTOR": INPUT NHS IF N25="" THEN NPS -N1\$ 60TO 80 75 NPS=N1S+"[SPC]&[SPC]"+N2S IF LENC NP\$>>37 THEN RUN 80 OPEN 3.4 B\$=CHR\$(16) 85 GUSUB 200 PRINT#3 PRINT#3 90 PRINT#3, CHR\$(14) "[CRSFD] **[SPC][SHIFTC] OMMODURECSPC364CSPC3##CSPC3CCRSRU3CSHIFTB3 95 PRINTWS PRINTWS, CHR#(15)"[CRS@U][11SPC] ---(SPC]:OFTWHRECSPC]---(SPC)(CRSRU] "B# 138ECOMMN1" 100 PRINT#3 FORF=1TO(38-(LEN(NP\$)+4))/2 PRINT#3,"[SPC]", NEXT 110 FOR T=1 TO LEN(NP\$)+4 PRINT#3,"[SHIFT@]
",:NEXT T PRINT#3 115 FORT=1T0(38-(LEN(NP\$)+4))/2 PRIN TW3, "[SPC]"; NEXT T 120 PRINTW3, "[SHIFTO][SPC]"NP#"[SPC] [SHIFTQ]"B\$"38[COMMN]" 125 FORT=1T0(38-(LEN(NP#)+4))/2 PRIN T#3,"[SPC]"; NEXT T 130 FOR T=1 TO LEN(NP\$)+4 PRINT#3,"[SHIFTQ] NEXT T PRINT#3 135 PRINT#3 PRINT#3."[CRSRD][75PC][SHIFTB] YESPOJ"NA#"[CRSRU]", B#"38[CDMMN]" -PR INTER 140 PRINT#3 PRINT#3,CHP#(14)"[CPSRD] ##[SPC1[SHIFTC]OMMUDORE[SPC]64[SPC]# #ESPECIERSRUJESHIFTBI"EHR#(15) 145 PRINT#3 GOSUB 200 150 PRINT#3, "ICRSRDIESPOIESHIFTCIARA [SPC]A [SPC][CRSRU]"N1\$ 155 PRINT#3, "CORSEDICSPOISSHIFTCIARA [SPC]B [SPC][CRSRU]"N2\$ 160 GOSUB 200 165 PRINT#3 PRINT#3, "CCRSRDJC7SPCJCSHIFTB] YESPOI"NHS"[CRSRU]"B\$"38[COMMN] 17G PRINT#3 PRINT#3 PRINT#3 GOSUB 20 180 RUN 200 FOR T=1 TO 39 PRINT#3, "-"; 'NEXT PRINT#3 RETURN

El Misterio de los Pokes (VI)



Por Diego ROMERO

He recibido bastantes sugerencias por teléfono y por correo sobre esta serie de artículos, y como toda la revista está abierta a opiniones constructivas, creo que tenéis razón en que es mejor publicar el mapa de memoria completo y luego ir analizando las distintas partes

Algunas de las cartas recibidas decian que no estaba siendo suficientemente detallado con las funciones de las posiciones de memoria, por ejemplo con la posibilidad de copiar una zona de memo-

ria en otra utilizando las posiciones de memoria 88-96 (publicado en magia en el número 6), la posibilidad de eliminar la interrogación en los inputs, etc

Creo que en mi defensa podría mencionar las cartas recibidas en sentido contrario (querían que fuese más rápido)

Como para poder contentar a todos o al menos a la mayoria debería hacer una encuesta y atenerme a los resultados, pero esto tardaria bastante y muchas veces las encuestas no son fiables porque los que se suelen quejar no las contestan pasamos a publicar el mapa de memoria del C-64, que fue publicado en Club Commodore, luego el del VIC, y quizás alguno de los PETs. Cuando terminemos con todos ellos, creo que podemos dedicar esta sección a comentar las posibilidades de las posiciones de memoria más útiles y populares sin tener que seguir el orden numérico como hasta ahora, mientras tanto estoy abierto a sugerencias (las prefiero por carta, por lavor)

Hex	Decimal	Description	Her	Decimal	Description
0000	0	chip directional register	0069 DB6E	105-110	Accum#2: Exponent, etc.
0001	1	Chip I/O; memory & tape control	D06F	111	Sign comparison, Accel vs #2
0003-0004	3-4	Float-Fixed vector	0070	112	Accumbl la-order (rounding)
0005-0004	5-6	Fixed-Float vector	0071-0072	113-114	Cassette buff len/Series pointer
0003-0000	7	Search character	00/3 00BA	115-130	CHRGET aubrowtine; get Basic char
		Scan-quotes flag	007A 007B	122 123	Basic pointer (within subrin)
0000	- ;	TAB column save	COSB COSF	139 143	RND seed value
0009	10	O-LOAD, 1-VERIFY	0090	144	Status word ST
DOOA		Input buffer pointer/# subscrpt	0091	145	Revoultch PIA: STOP and NV5 [lags
0000	11	Default DIM flag	0092	146	Timing constant for tape
0000	12	Type FF-string, OD-numeric	0093	147	Load-O, Verlfy+1
000D	13	Type #0-integer, 00-floating point	0094	148	Serial output: deferred char flag
2000	14	Type adelitedet, overloading point	0095	149	Serial deferred character
000#	15	DATA scan/List quote/seary flag	0096	150	Tape EOT received
0010	16	Subscript/FHx []ag	0097	151	Register save
1100	17	0-1MPUT; \$40-GET, S98-READ	0098	152	How many open files
0012	10	ATM mign/Comparison eval flag	0099	153	Input device, normally 0
0013	19	Current I/O prompt flag	009A	154	Output CMD device, normally 3
0014-0015	20-21	Integer value	0098	155	Tape character parity
0016	22	Pointer, temporary strg stack	0090	156	Byte-received flag
0017 0016	23 24	last temp string vector		157	Direct=S80/HUM=0 output control
0019 0021	25-33	Stack for temporary strings	0090	158	7p Pass 1 error log/char buffet
0022 0025	34 37	Utility pointer area	009E		To Pass 2 err log corrected
0035 005V	38 42	Product area for multiplication	009F	159	
002B 002C	43 44	Pointer Start of Basic	00A0 00A2	160 162	JITTY Clock HML
002D 002E	45 46	Pointer. Start-of-Variables	EAGO	163	Secial bit count/EO1 flag
002F 0010	47 48	Pointer Start of Acrays	DDA4	164	Cycle count
0031 0032	49 50	Painter End of-Arrays	00A5	165	Countdown, tape write/bit count
DO33 0034	51 52	Pointer String storage (moving down)	00A6	166	Tape buffer pointer
0015 0036	53-54	Utility string pointer	00A7	167	Tp Wrt ldr count/Rd pass/inbit
	55-56	Painter: Limit of memory	BAGG	168	Tp Wrt new byte/Rd error/inbit cni
0037-0038	57-5B	Current Besic line number	00A9	169	Wet start bit/Rd bit err/stbit
0039-003A	59-60	Previous Basic line number	OGAA	176	Tp Scan;Cnt;Ld,End/byte assy
DO 38-003C	61 62	Pointer Basic statement for CONT	DOAB	171	Wr lead length/Rd checksom/parity
003D 003E	63 64	Current DATA line number	OOAC - BOAD	172 173	Pointer: tape bufr, scrolling
DO3F 0040	65 66	Current DATA address	TAGG SAGG	174 175	Tape and adds/End of program
0041 0042		Input vectof	00B0 00B1	176 177	Tape timing constants
8043 0044	67 68	Current variable ness	0082 0083	178 179	Pote: Start of tape buffer
0045-0046		Current verlable address	0084	180	leTp timer enabled; bit count
0047 0049	21-72	Variable pointer for FOR/NEXT	0085	161	Tp EOT/RS232 nest bit to mend
0049 004A	73 74	Y save, op mave, Basic pointer save	0086	182	Read character error/outbyte buf
004B-004C	75 76	Comparison symbol accumulator	00B7	183	I characters in file name
004D	77		0088	184	Current logical file
004E-0053	76-83	Misc work area, pointers, etc.	0089	185	Current secody address
0054 0056	B4-86	Jump vector for functions	DOBA	186	Current device
0057 0060	87 96	Misc numeric work area	2800 8800	187 188	Pointer to file name
0061	97	Accum@i: Exponent	GOBD	189	Wr shift word/Rd Input char
0062-0065	90-101	Accumel: Mantissa	COBE	190	# blocks remaining to Wr/Rd
0066	102	Accumel Sign	0085	191	Serial word buffer
9067	101	Series evaluation constant pointer	0000	192	Tape motor interlock
0068	104	Accumbl hi-order (overflow)	0001 0002	191-194	1/O start address
5000			0002 0012		***

```
Decimal Description
Stex
                                                                                                                                                 Max.
                                                                                                                                                                        Secimal Description
                                                                                                                                                EDDG-FFFF 57144-65515 ROM. Operating System EDDG FFFF 57344 65535 Alternate: RAM FF81 FFF5 65409-65525 Jump Table, Including: FFC6 - Set Input channel FFCC - Restore default 1/0 channels FFCC - IMPUT FFD2 - PRIMT FFD2 - PRIMT FFD2 - FFST Stop key FFE4 - GET
                                             Kernel metup pointer
Last key pressed
# charm in keybd buffer
Screen reverse flag
 0003-0004
                          195-196
 0005
                                             # charm in keybd buffer
Screen reverse flag
Find of line for input pointer
Input cursor log (row, column)
Which key: 64 if no key
O*flash cursor
Cursor timing countdown
Character under cursor
Cursor in blink phase
Input from screen/from keyboard
Pointer to screen line
Position of cursor on above line
O*direct cursor, slae programmed
Current screen line length
Row where cursor lives
Last inkey/checksum/buffer
f of INSERTS outstanding
Screen line link table
Screen color pointer
Reyboard pointer
RS 232 Rcv pntr
RS 232 Tz pntr
Floating to ASCII work area
 0007
                          199
  00CB
                          201 202
 open soca
                          203
204
 DOCC
 ence
                          205
                          206
 DOCE
 0000
                          201
                          209-210
 00D1-00D2
 ann's
 cop5
 0.006
 0007
 DODA
                         217-242
243-244
245 246
247-248
 00D9 00F2
00F3 00F4
00F5 00F6
00F7-00F8
                          249-250
256-266
256-318
 0079 00FA
                                              Tape error log
Processor stack area
 0100-303E
0100-01FF
                          256-511
512-600
                                                          Input buffer
 0200-0258
                                              Logical file table
Device & table
Sec Adds table
                          601-610
  0263-0260
                          611 620
                          621-630
61, 640
641 642
 0260-0276
                                                                                                                                                 A000, BON control vectors
                                              Sec Ands table
Keybd buffer
Start of Rasic Memory
Top of Basic Memory
Seriel bus timeout flag
Current color code
                                                                                                                                                 AGOC, Reyword action vectors
 0281-0787
0283-0284
                                                                                                                                                  AUS2, Function vectors
                          641 644
645
                                                                                                                                                 AUSO, Operator vectors
 0785
                          646
                                                                                                                                                  AD9E, Reywords
                                               Color under cursor
                                              Color under cursof
Screen memory page
Mam mize of keybd buffer
Repeat all keys
Repeat speed rounter
Reyboard Shift/Control flag
Lamt shift pattern
Reyboard table metup pointer
Reyboard shift mode
Ovscroll enable
  0287
                                                                                                                                                  A19E, Error messages
  0288
                          648
                                                                                                                                                  A328, Error message vectors
  0.289
                                                                                                                                                  A365, Hisc messages
  028A
                          650
                                                                                                                                                  A38A, Scan stack for FOR/GOSUB
  0288
 02BC
02BD
                          652
                                                                                                                                                  A388, Move memory
                                                                                                                                                  A3fB, theck stack depth
  0285
                                                                                                                                                 A408, Cneck memory space
A435, "out of meanry"
                          655-656
  028F 0290
  0291
                          658
                                                                                                                                                  Au37, Error routine
                                              RS-232 control reg
RS 732 command reg
Bit timing
RS-232 status
bits to send
RS-232 speed/code
                          659
  0293
                                                                                                                                                 A469, BREAK entry
                          66H
661-662
                                                                                                                                                 A-74, "ready.
A480, Ready for Bests
                                                                                                                                                                 · ready .
  0295-0296
                          663
  D297
                          664
                                                                                                                                                  A496. Handle new line
  0299-029A
                          665
                                              RS-212 speed/code
RS-212 receive pointer
RS-212 input pointer
RS-212 transmit pointer
RS-212 output pointer
RS-212 output pointer
CIA 2 [NM1] Interrupt Control
CIA 1 Timer A control log
                                                                                                                                                  A533, Re-chain lines
                          667
                          668
                                                                                                                                                  A:60, Receive input line
 0290
                          669
                                                                                                                                                  AS79, Urunch tokens
  0792
                                                                                                                                                  A6.3, Fand Basic line
  0297 0284
                          671-672
                                            CIA 2 (NMI) Interrupt Control
CIA 1 Timer A control log
CIA 1 Interrupt Log
CIA 1 Timer A enabled flag
Screen row marker
(Sprite 11)
Cror message link
Basic warm start link
Crunch Basic tokens link
Print tokens link
Start new Bamic code link
Get arithmetic element link
SYS A-reg save
SYS T-reg save
SYS T-reg save
SYS T-reg save
USR function jump
Break interrupt vector
NMI Interrupt vector
CPFN vector
CLOSE vector
Est-output vector
Restore I/O vector
(F334)
Restore I/O vector
(F333)
INPUT vector
UTDE
(F157)
INPUT vector
(F157)
(F16A)
                                                                                                                                                  A6-2, Perform [NEW]
                                                                                                                                                  Aobb, Pertorm [GLR]
  02A2
 02A3
                                                                                                                                                  A68L, Back up test pointer
                                                                                                                                                  A69c, Perform [LIST]
  07A5
                          704 766
768 769
770-771
 03C0-03PE
                                                                                                                                                  A742, Perform [FUR]
                                                                                                                                                  A7ED, Execute statement
 0302-0303
 0304-0305
                          772-773
                                                                                                                                                  ASID. Perform (RESTORE)
                                                                                                                                                  ABZG, Break
 0108-0309
                          776-777
                                                                                                                                                  A82F, Perform [STOP]
                                                                                                                                                 ABJI, Perform [END]
AB57, Perform [CONT]
 0300
                          780
 9 100
                          781
                                                                                                                                                  A871, Perform [RUN]
 030F
                                                                                                                                                 A883, Perform [GOSUS]
A8AO, Perform [GOTO]
                          784 785
 0110-0312
 0314 0315
                         788 789
790 791
 0316-0317
                                                                                                                                                  ASD2, Perform [RETURN]
                          792 793
                                                                                                                                                  ABFB, Perince [DATA]
                          794 795
 0318-0318
                                                                                                                                                 A936, Scan for next statement A92d, Perform [1F]
 011E-031F
 0320-0321
                         800-801
                                                                                                                                                  A938, Perform [RES]
                         802 801
 0322-0323
                                                                                                    (F157)
                                              IMPUT vector
Output vector
Test-STOP vector
                                                                                                                                                  A948, Perform [ON]
0124-0325
0126-0327
                         804 605
                                                                                                                                                  A968, Get fixed point number
                                                                                                     (F6ED)
 0326 0329
                         808 809
                                                                                                                                                  A9A5, Perform [LET]
                                             GET vector
Abort 1/0 vector
                                                                                                    (F13E)
 032A-0328
                         810 811
                                                                                                                                                  AABO, Perform [PRINT#]
                         812 813
814 815
816 817
                                                                                                    (F32F)
0320-0370
                                                                                                    [PE66]
                                             Warm start vector
LOAD link
SAVE link
                                                                                                                                                  AA86, Perform [CHD]
                                                                                                    LF (A5)
0330-0331
                                                                                                                                                  AAAO, Perform [PRINT]
0332-0333
033C 03FB
                         818 819
828-1019
                                                                                                                                                 ABIE. Print string from (y.a)
AB3B, Print forant character
AB4D, Bad input routine
                                             Cassette buffer
[Sprite 13]
(Sprite 14)
                         837 894
896 958
0340-0376
 0380 03BE
0380 038E 896 958 (Sprike 18)
0300 03FE 950 1022 (Sprike 15)
0400 07FF 1024 2047 Screen memory
0800-9FFF 2048 40959 Basic ROM memory
8000 9FFF 40950 49351 ROM 88051C
ADDO 8FFF 40950 49351 ROM 8831C
ADDO 8FFF 40950 49351 ROM 8831C
ADDO 8FFF 40950 49351 ROM 8831C
ADDO BFFF 40950 54351 Alternate: RAM
0000 CFFF 49352-53247 RAM memory, including alternates: RAM 8851C
ADDO DOZE 51748 51294 Video Chip (6566)
                                                                                                                                                  AB7B, Perform [GET]
                                                                                                                                                 ABA5, Perform [INPUT#]
ABBF, Perform [INPUT]
                                                                                                                                                  ABF9, Prompt & input
ADDO BFFF 49160 59151 Alternate: RAM
COGO-CFFF 49152-53247 RAM memory, including alternate
DOOD DOZE 51748 51794 Video Chip (6566)
D400 D41C 54272-54300 Sound Chip (6568) SID)
D800 DBFF 55296 56319 Color nybble memory
DC00 DC07 56320 56335 Interface chip 1, IRQ (6526 CIA)
DD00 DD07 56576-56591 Interface chip 2, MMI (6526 CIA)
DD00 DFFF 53248-53294 Alternate: Character set
                                                                                                                                                  ACOS, Perform [READ]
                                                                                                                                                  ACFC, Input error messages
                                                                                                                                                                                                              (continuarà)
```



lagia

La MAGIA son trucos, la MAGIA es divertida.

La MAGIA es hacer lo que nadie se ha atrevido y resulta ser la fuente más completa de información para la informática práctica.

La MAGIA es una sección mensual llena de consejos, trucos, de esto y aquello del mundo del software, hardware y aplicaciones, trucos descubiertos por los demás que hacen que la informática sea más fácil, más divertida o más animada.

MAGIA habla de ideas sencillas, programas de una sola línea, subrutinas útiles, hechos de informática poco conocidos y otras cosas de interés.

Los trucos de magia enviárnoslos comprobados, pues hay varios incorrectos. ;Ah! y no nos mandéis trucos repetidos... ¡Listos!. Premios: Ver pág. 5.

DESCRIPTION AS TO EFROCESO

Dada la gran cantidad de usuarios de los Microordenadores aplica dos a la radio RTTY/CW, os envío un corto pero eficaz programa para iniciaros en la Modalidad ASCII, con el cual podréis transmitir y recibir todos los caracteres Ascu de vuestro Ordenador, o sea que todo lo que hagáis vosotros en vuestra pantalla le saidrá al correspondal, a diferencia del código BAUDOT que queda limitado a emitir letras núme ros y poco mas

Con este programa que es el primero de una serie que posteriormente

publicaré queda la puerta abierta para el teleproceso Envio y recepción de listados

Acceso a programas de otros ordenadores

En sin todo cuanto os quiera dar la fantasia de los ordenadores

aplicados a la radio

Como es lógico para que el programa funcione al 100×100 el corresponsal tiene que tener otro ordenador igual sobre todo en lo que res pecta a color etc., no sé cómo funcionaria teniendo el corresponsal otro ordenador distinto, yo las pruebas las he realizado con ordenadores

La eficacia en la transmisión también depende mucho del Modem a utilizar, yo las pruebas las he realizado con un Modem autoconstruido por mi y el resultado ha sido bastante satisfactorio incluso a 110 Baudios con señales en FM menores de UN Santiago de Smiter

Animo y a darle a las teclas!

Arturo Sánchez Galán Solosancho (Avila, EA I BDD

PROGRAMA EMISION RECEPCION "ASCIF

10 OPEN 2.2.3, CHR\$ (129)+CHR\$(192) 20 PRINT*(cir) EMISION 25 GET AS. IF AS=" 1" IHLN 100

30 PRINT AS.

40 PRINT #2,A\$.

50 GOTO 25

100 PRINT "(clr) RECEPCION" 110 GET AS. IF AS ="f" THEN 20

120 GET#2,A\$. PRINT A\$.

130 GOTO 110

La velocidad en baudios puede aumentarse sumando uno al primer CHR\$

50 Baudios CHR\$ (129) 75 Baudios = CHR\$ (130)

110 Baudios = CHR\$ (131)

No es recomendable pasar de una velocidad de 110 baudios a no ser que las señales de la otra estación sean muy limpias

MERGE-64

Cuando en el C-64 se efectúa un LOAD automaticamente se borra el programa que hasta ese momento estuviese en la memoria RAM Esto impide enlazar dos o más programas previamente almacenados en cinta. En otros ordenadores se dispone de la instrucción MERGE que permite realizarlo. Los usuarios del C-64 pueden simular esta instrucción de la siguiente forma

— Hacer un LOAD del primer programa
— Mover el puntero de inicio del BASIC de forma que apunte al final del programa que acabamos de cargar, para ello teclear POKE 43. PEEK(45) - 2: POKE 44, PEEK(46)

-Hacer un LOAD del segundo programa

-Volver a colocar el puntero de inicio del BASIC en la posición \$

POKE 43, 1: POKE 44.8

Si después de todo esto haceis un LIST, vereis que los dos programas aparecen enlazados: el programa que se cargó en 2º lugar comienza al final del primero

Conviene que el programa que se vaya a cargar en 2º lugar tenga numeros de lineas superiores a los del programa que se encuentre en memoria ya que el orden de los números de línea no se restablece (0)0 con los GOTO y GOSUB')"

Rafael A García Segura Montes de Oca, 9 - 2º D 29007 MALAGA

MERGE VIC-20

Os envío una subrutina que algunos ordenadores poseen pero no el

VIC-20, se trata de la función "MERGI"

La función "MERGE" sirve para unir dos programas en uno solo. El procedimiento es el siguiente: Se comprueba primero que entre los dos programas a unir no exista ningún número de línea idéntico, ya que en este caso la línea se perderia al ser el número igual. Luego se teclca en modo directo POKE 43, PEEK (45)-2: POKE 44, PEEK (46), y se carga el segundo programa con LOAD. Cuando éste se haya cargado se teclea en modo directo POKE 43,1: POKE 44,16, y ya se pueden listar los dos programas unidos.

Andres Portas Afenjar C/Rios Rosas, 5-49 D 29007 MALAGA

QUINIELAS

Poseo un CBM 64 desde hace 4 meses, y desde entonces estoy suscrito a vuestra revista Nº 3332.

Hace muy pocos dias adquiri la impresora MP8 80V, y os mando un pequeño programa para rellenar quinielas con esta impresora.

Os mando el listado y un ejemplo práctico en papel, de la impresora, Por supuesto es totalmente aleatorio, pero podría ser la de la próxima semana (por si acaso copiarla).

```
5 REM 00 QUINIELR 00
7 REM 00 JOSE A, FERNANDEZ VILLORIA.º
18 OPENI.4
28 PRINT#1,CHR#(14)CHR#(16)"25GUINIELR"
21 PRINT#1,CHR#(18)CHR#(16)"25 "
25 PRINT#1,CHR#(15)
     FORH-17060 PRINTOL, "-" : NEXTH
35 PRINTO1.CHR$(10)
40 FOR1=17014
42 IF1C10THENPRINT01," ";
45 PRINT01,1;" PARTIDO",
30 FOR = 1100
60 W=INY(RNIC0)#11)'IFM<=5THENS#=" 1
78 IFM<=6ANDWJSTHENS#=" % "
80 IFWJOTHENS#=" 2 "
    PRINTOL, CHRS(16)368,
90 PRINTO
110 NEXTJ
111 PRINTOL
112 NEXTI
113 PRINTO1, CHR6(16) "65 VILLORIA."
114 FORH=1TO68 PRINTO1."-", NEXTH
115 PRINTEL
120 CLOSE1
```

READY.

QUINIELA

1234	PARTIDO PARTIDO PARTIDO PARTIDO	× 2	1 1 2	1 X 1	1 1 1	1 1 2	×	1 1	1 1 1 1
5	PARTIDO	×	×	×	i	1	×	î	. 2
- 6	PARTIDO		×	1	. 5	i	×	. 2	- 2
7	PARTIDO	1	14	1	2	1	34	1 -	1
	PARTIDO	31	- 34	2	2	2	2	1	1
9	PARTIDO	24	1	1	1	14	30	1	1
18	PARTIDO	1	×	M	1	1	1	30	2
- 11	PARTIDO	5	×	2	1	1	1	- 1	1
12	PARTIDO	ж	1	×	2	1	1.	1	34
13	PARTIDO	3	1	4	3	30	1.	1	2
14	PARTIDO	1	1.	1	1	14	1	1	2

José Antonio Fernandez Villoria C/ García nº 5, 2º Izda Guón (Asturias)

SALVASUANDA DE MEMORIA

Queridos amigos de COMMODORE WORLD; es esta la primera vez que envío una colaboración a la revista, aunque soy suscriptor desde el nº 1, a ver ai entro en el tinglado de los premios y regalos, ya que observo que para los sucriptores que se "quedan en casa" no hay nada Bien, paso a comentar mi "trabajillo"

Me voy a referir a un artículo que con el título "SALVAGUARDA DE MEMORIA EN PERIFERICO", apareció en la revista nº 6, firmado por Rafael Jiménez Jiménez, ya que es un programa interesante para todos aquellos que les guste programar en código máquina, toda vez que permite salvar las rutinas de una forma más rápida y ortodoxa que el encerrarlas en BASIC, amén de la facilidad de carga, ya que aunque tengamos otro programa en memoria, podemos cargar la rutina, sin afectar al programa existente, lo que no sería posible si la rutina estuviera en BASIC

Pues bien, mis comentarios son los siguientes;

1º,-El programa que nos presenta nuestro amigo Rafael no funciona correctamente, ya que no salva el contenido de la dirección dada como dirección final a la pregunta que hace el programa. Se puede comprobar salvando una zona de memoria, se borra posteriormente, y se carga lo salvado, se observa que falta el contenido de la última dirección. La causa está en que la rutina SAVE del Kernal, no salva el contenido de la dirección especificada como final. El defecto del programa es importante, toda vez que las rutinas en código maquina, suelen terminar con la instrucción RETURN, y si esta falta, la rutina no va a funcionar, corriendo el peligro de causar un "atasco" al ordenador al lanzar la rutina. El problema se soluciona sustituyendo la linea 70 por la siguiente:

70 PRINT: INPUT"DIR, FINAL DE MEMORIA", DY DY-DY+1. GOSUB3000; LF BB MF-BA

Con esto el problema queda solucionado, ya que el programa suma i a la dirección final

2º.-Cuando una rutina salvada con este programa, se carga en

modo directo, dicha carga afecta a ciertos punteros de página cero, de tal manera que si en este momento tenemos un programa BASIC en memoria, existen pocas posibilidades de que después de la carga de la rutina funcione, a menos que tomemos algunas precauciones antes de cargar la rutina. La causa de esto, tenemos que buscarla de nuevo en las rutinas del Kernal, en este caso en la rutina LOAD. Esta rutina, devuelve la última dirección cargada, dirección que utiliza el INTER-PRETE BASIC para saber dónde termina el programa, y dónde por lo tanto puede almacenar las variables, para ello, almacena esta dirección devuelta por la rutina LOAD en los punteros de página cero \$2D y \$2E. Si la rutina cargada reside fuera de la zona de memoria BASIC, lo cual es bastante frecuente ya que la zona de memoria normalmente utilizada para las rutinas en código máquina esta entre 49152 y 53247, muy por encima de la zona BASIC, los punteros de la página cero se actualizan a un valor incorrecto, lo que ocasiona que el INTERPRETE BASIC emita el mensaje "OUT OF MEMORY", no permitiendo hacer nada a menos que se haga NEW, con lo que perdemos el programa.

Existen dos soluciones para este problema, la primera consiste en salvar los punteros de página cero, a continuación cargar la rutina, y posteriormente restaurar los punteros de nuevo; la segunda consiste en cargar la rutina con un LOAD en modo programa, en cuyo caso el INTERPRETE BASIC, no actualiza los punteros. A continuación se

describen los dos métodos

PRIMERA SOLUCION.—Cómo salvar y restaurar los punteros.

1º.—Hacer POKE1020, PEEK (45), POKE1021, PEEK (46) (RETURN)

2º.—Hacer LOAD"NOMBRE", NUMERO DE PERIFERICO, 1 (RETURN)

3º.—Esperar que se cargue la rutina.

4º.—Hacer POKE45, PEEK (1020) POKE46, PEEK (1021) (RETURN) La rutina se habrá cargado sin afectar al programa BASIC SEGUNDA SOLUCIÓN.—Carga de la rutina en modo programa Es necesario para ello añadir un par de lineas al programa BASIC de la siguiente forma:

2 LOAD "NOMBRE", NUMERO DE PERIFERICO, 1

Hacer RUN 2 (RETURN), y esperar a que se cargue la rutina, enton-ces borrar las lineas 1 y 2, con lo que tendremos cargada la rutina, y el

programa BASIC correcto.

ya, para finalizar, ahi está mi propia versión del programa. No se utilizan las rutinas preparatorias SETNAM y SETLFS, ya que lo único que estas rutinas hacen es colocar datos en la página cero para su posterior utilización por la rutina SAVE. En este programa, el trabajo de esas rutinas lo realiza el BASIC directamente, con lo que el programa se hace mucho más corto, aunque es necesario entrar con un SYS en la dirección adecuada de la rutina SAVE, esto se hace al final de la linea 60. El programa comprueba también que la grabación ha sido correcta

5 REM *** MEMORIA A PERIFERICO *** 6 REM

10 PRINT"[CLR][2CRSRD]" - INPUT"NUMBRE [SPC]";N\$ N=LEN(N\$):IFN>10THEN N\$=LE FT\$(N\$,10):N=10

20 INPUT"[CRSRD]DIR.[SPC]INICIAL[SPC] ";D:POKE194,INT(D/256);POKE193,D-INT (D/256)#256

30 INPUT"[CRSRD]DIR.[SPC]FINAL[SPC]"

, D · D≠D+1 : POKE175, INT(D/256) · POKE174, B-INT(D/256)#256

40 INPUT"[CRSRID11-CHSS.[SPC]8-DISCO"

:A:IFAC>IANDAC>860T040

50 POKE186,A:POKE183,N:POKE184,1:POK E185,1:POKE187,192.POKE188,2

60 FORJ#1TO N:POKE703+J/ASC(MID\$(N\$/

J, 1)): NEXT: SYS62957

70 PRINT"[2CRSRD]REBOBINA[SPC]LA[SPC] CINTAESPOJYESPOJPULSAESPOJUNAESPOJTE CLA":PUKE198,0:WHIT198,1

80 PRINT"[CLR]VERIFY"CHR\$(34)N\$CHR\$(34)", "8", 1"

90 POKE198,2:POKE631,19:PÚKE632,13:E ND

Por ultimo, aunque la sintaxis correcta de carga es. LOAD "NOMBRE", NUMERO DE PERIFERICO, I

Si se utiliza como periférico el cassette, es suficiente hacer LOAD, sin ningún otro dato para la carga de las rutinas

Alfredo Pérez Pérez Bolarque - GUADALAJARA



TECLAS EXTRAÑAS



Me gustaria que me contestaran a un par de dudas que tengo sobre mi VIC-20 En primer lugar quisiera saber cuál es la funcion de tres

teclas de la consola del ordenador, que son las

signientes. FMILIO GRANES FERNANDEZ

C/PLAZA DE LA CARRETERIA, Nº 33

CONSTANTINA (SEVILLA)

Las teclas " 4", "2" y "@" no tienen otra función más que la de servir de símbolos gráficos, aunque algunas de ellas, en determinados programas (ej. "WEDGE" con la unidad de discos) tengan una función especial.

GRAFICOS EN ROM



Lo primero felicitaros por vuestra revista, especialmente por la sección de magia que tantos trucos útiles aporta

Mi consulta va sobre los caracteres progra-

Tras haber leido la sección del manual dedicada a esto, tantas veces como para gastar las letras, he conseguido enterarme de cómo fun-

VIC-20	Ptas
Farjeta 64K RAM+2K EPROM Farjeta 2K+2K F-PROM Farjeta 40/80 columnas+	18,900 3,695
3.5K RAM	16 350
Unidad de expansión de 2 slots	3,200
Unidad de expansión de 5 slots	9 900
Printer Buffer PAR/PAR 16K	24 790
CBM64	
farjeta 80 columnas	17 90u
Tarieta 4K+8K EPROM	4,750
Unidad de expansión de 2 slots	3,200
Unidad de expansión de 5 slots	12.995
Printer Buffer SER/PAR 32K	27 950
PARA VIC-20 Y CBM64	

lableta grafica KOALA-PAD 23 500 PARA TODO TIPO DE ORDENADORES

PROGRAMADOR DE EPROMS 16.995

BORRADOR DE EPROMS 8.775

Pedidos e informacion:

DELTABLE Colón, 20-4ª SILLA (Valencia) Teléf.: (96) 120 29 25

Pero ahora que ya estaba contento me he encontrado con el problema (para mi hasta ahora insalvable) de que no sé (ni lo dice el manual) cómo copiar los caracteres del set 2 es decir mayúsculas y minusculas

Otra duda sobre la cual no he encontrado respuesta en ninguna revista o libro es de, que hay que hacer para que un programa que se esté cargando desde el cassette o la unidad de discos se autoejecute

Me despido no sin antes felicitaros nuevamente por vuestra/nuestra revista

Un asiduo lector, MIGLEL F. BAEZ DE AGUILAR G (MALAGA)

caracteres de la ROM a la RAM

Las minusculas, inversas, etc., están situadas en las ROM después de los caracteres en mayus-

53248	\$D000	Mayúsculas
53760	\$D200	Gráficos.
54272	\$D400	Mayúse, invertidas
54784	\$D600	Gráficos invertidos.
55296	\$D800	Minusculas
55808	\$D 400	Mayusculas y gráfic
56320	\$D C00	Minusculus inv
56832	\$DE 00	Minusculas y gráf. i

A lo del auto-run, lo publicaremos en próximos números pues hemos recibido numerosas cartas al respecto.

COMPAYIBILIDAD DE CINTAS



Hola, os escribo porque tengo un problema con mi VIC-20 y si podeis, quiero que me lo resolvais

le la posibilidad de intercambiar programas con VIC-iosos Ingleses, pero he oído que sus sistemas de televisión son PAL (o NTSC), y el nuestro es el de 625 lineas. Funciona una cinta con programas ingleses en un VIC español y viceversa

JAVIER RECIO MATA C/FERNAN CABALLERO, 20 41001 SEVILLA

En primer lugar la contestación rápida a tus dudas, las cintas utilizan el mismo sistema en todos los VIC-20 del mundo. En segundo lugar te indicaré que el sistema Inglés es el PAL, que es el mismo que el nuestro. La diferencia entre sus VIC-20's y los nuestros es que el canal de sonido del sistema PAL británico utiliza una subportadora de audio de frecuencia superior a la nuestra (6.0 MHZ en lugar de 5,5 MHz), esa es la causa de que los ordenadores traidos de Inglaterra no se oigan bien en nuestros televisores, y si ajustas la sintonia para oírios bien entonces no lo ves. La solución es un ajuste que tiene el modulador (esa capta que se enchufa al televisor). El caso es el mismo con los C-64 de las Canarias y que son extranjeros (hay que retocar un tornillito). Algunas veces hemos oído que tenían que cambiarles un carisimo circuito integrado, pero esta es una idea incorrecta.

GOSURS ENCADENADOS



Poseo un VIC-20 desde primeros de este año y considero su labor muy interesante. Desde el punto de vista de un VIC-ioso su

revista lleva corto el contenido para este ordenador aunque comprendo que no se puede abandonar a los pobres propietarios de C-64, 4000, 8000, etc.

En el MANUAL DEL USUARIO se comenta que después de un GOSUB, la instrucción RETURN devuelve el programa a la instrucción siguiente al GOSUB, pero ¿qué ocurre cuando hay dos GOSUBs encadenados? es lo mismo que con las instrucciones FOR . NEXT

Efectivamente, los GOSUBS pueden encade-narse como si fuesen FOR...NEXT, pueden llegar a encadenarse hasta 24 subrutinas unas dentro de otras; si se sobrepasa este número, el ordenador contesta con ?out of memory. Para comprobarlo basta con escribir... 1 GOSUB I y hacer Rt N

ROWING LA BOM



CARTA BLANCA

Quisiera hacer unas cuantas preguntas, de las cuales no he encontrado respuesta en ningun sitio

a) Al copiar la ROM en la RAM chay algun método para copiar el interprete del BASIC en cinta de cassette

b) Este programa, por lo tanto, podria hacerse correr en otro microordenador con la misma CPU⁹

c) Si asi fuera ¿este otro ordenador se podria hacer funcionar con los mismos programas que el primero?

d) Y por ultimo, los cartuchos enchufables Sustituyen a la ROM? y si es así ¿ocupa las mismas posiciones de memoria?, ¿estos se pueden copiar en la RAM?

Os habré dejado aturdidos ano es así! Se despide de vosotros rápidamente IVO PLANA VALLI F

C/CAPUTXINS, 30, 4, 1° **TARRAGONA**

A .- Puedes salvario con un monitor de lenguaje maquina o con el programa aparecido en el número 6 (pág. 46) "Salvaguarda de memoria en periféricos

 \vec{B} — Si, pero el mapa de memoria de este otro microordenador tiene que ser igual que la del C-64, es decir, o es un C-64 o una copia idéntica C.—Claro, sería un C-64 o su "fotocopia"

D.—Algunos la sustituyen, otros se suman a ella sin sustituirla, etc. Todo depende de la combinación de las señales EXRON, GAME, en el Port de expansión y el contenido de la dirección I sla tienes en los cuadros de "El Misterio de los Pokes" del número 4)

COMPRA DE PITOERAMAS



He comprado el número 5 de vuestra revista para informarme mejor sobre las posibilidades de el ordenador personal Commodore-64 pues-

to que pienso adquirir en breve uno. Al leer vuestra revista he visto con asombro la calidad y clandad de esta, que tratando temas con bastante profundidad se hace entender por personas que, como yo, no tenemos un nivel de conocimientos de informática demasiado elevado. También he leido algo que me ha hamado poderosamente la atención y es un bole-tín llamado "Servicio de Cintas", en el que, según creo entender, ofrece un servicio de venta en cintas magnetofónicas de programas "publicados" al precio único de 850 pesetas más 75 por gastos de envio. Es este el punto que os pido me aclareis, el de los programas "publicados", o sea, si se puede comprar cualquier programa, ya sea de juegos o de gestión, por muy poco nombrado que esté en la revista. al precio antes mencionado MİGLEL NUÑEZ PEREYRA

LANZAROTE Se puede comprar cualquier programa de los que aparecen listados en la revista, no los programas a los que hace referencia la publicidad.

... Y

SEAMOS

FALLOS EN LOS PEGGEAMAS



En primer lugar, queria felicitaros por vuestra revista, quizás un poco técnica, pero muy completa. Tengo un C-64, y acabo de adqui-

tir el cassette, me ha ocurrido, que al copiar algunos programas, he encontrado fallos, que no he podido resolver, ni aún mirando las meteduras de pata de los siguientes numeros

Los fallos son los siguientes

-En el nº 3, dando valor a las teclas de funciones, el ordenador me da un Illegal

quantity en la linea 1030

—En el mismo nº 3, el programa "Mosqueando al mosquito", me da un error también, debido a que algun POKE maidito me ha borrado todos los signos más, menos, comillas, comas, etc... y además no puedo volver a escribirios

—Por último, en el manual, el programa de "teclado de piano" que hay al final, me da lllegal quantity en la línea 310. Tanto este error, como el de las teclas de funciones, son debidos a una secuencia ASC, la cual, aun no he podido descifrar

ENRIQUE LAHUERTA PEREZ C/HERMANOS IBARRA, 6-8

HOTEL PATRIA, SOOOT ZARAGOZA

1.—El error puede que no esté en esa linea, sino en alguna otra, en la que se definen (F\$(1) o I con unos valores que hacen tiegal su utilización en la linea 1030

2.—Si haces RUN sin haber salvado el programa antes, te arriesgas a que algún POKE ravillo te bloquee el ordenador o te haga casas que no debiera; si esto sucede, desconectar el ordenador y vuelves a cargar el programa que habtas salvado

3.—Pasa lo mismo que can la primera pregunta, en este caso verifica la linea 300.

INVERSION EN DATAS



Me gustaria saber si se pueden leer los datos de las sentencias DAFA de un programa directamente en orden inverso al que tienen

Insisto en lo de "directamente", ya que el problema en si tiene fácil solución metiendo previamente los datos en una matriz o haciendo un RESTORE y leyendo en cada vuelta un dato menos

FRANCISCO SAEZ SOTO C/VIRGEN DEL ROSARIO, 28 4LCANTARILLA (MURCIA)

En lugar de usar el típico lazo FOR Z=1 TO N. READ A: NEXT, sustitúyelo por el que la hace en orden inverso.

FOR Z N TO 1 STEP -1, RE 1D A: NEXT Es una solución sencilla a tu gran problema no?

EQUIVALENCIAS



¿A qué dirección del COM-MODORE-64 equivalen las siguientes, que pertenecen al VIC-20 y se entran por POK1.

POKE 30208... y POKE 30720. (esta ultima ta podéis ver en el número 6 de la revista, en la pág. 39 y dentro del prg. "UFO" en la línea an

Una idea: podriais explicar a qué direcciones del C-64 equivalen las direcciones del VIC-20 ampliado (aparte de lo que apareció en la

revista número 2), en este caso, por ejemplo, POKE 30000. . y pico

 —Cuando comienzo a hacer un programa y pongo la instruccion

10 PRINT" [SHIFT] [CLR]" y después listo el prg. algunas veces (últimamente bastantes) esa linea no aparece impresa

A que se puede deber esto"

Fallos frustrados del manual del usuario del C-64. El programa "teclado de piano" de la pág 147 del manual de usuario del C-64 no hace nada

1. La 30720 no es propiamente la dirección sino la que se SUMA a 7680 (la memoria de pantalla) para que dé 38400 (la memoria de color)

2. Que no aparezca una linea de programa, habiéndola introducido pulsando RETURN puede porque en vez de RETURN hayas pulsado SHIFT/RETURN, que produce el mismo efecto y no guarda la linea en la memoria, o que si estás utilizando cartuchos de ampliación (o el superexpander), que este tenga algún fallo o recalentamiento

THE AS DE TUNCTON



Quisiera me dijesen como se pueden programar las teclas de funciones especiales pues en el manual del VIC-20 se hace alusion a ellos pero no

pone cómo programarlos

Espero que mi carta no haya sido larga pero es la primera vez que escribo a la revista

Me despido de ustedes contando con su colaboración

FRNESTO SARRALDE LOPEZ C/PINTOR VERA FAJARDO, Nº 18- 3°C 01008 VITORIA-GASTEIZ 4LAVA

Las teclas de funciones pueden ser programadas, en el numero 3 se explica cómo se hace.

INSTRUCCION BOTO



Como la mavoria de los lectores estov metido en el mundo de la informática, quizas un poquito más prolundamente puesto que para

mi es el sustento de mi lamilia. Hago referencia a la carta firmada por Guillermo Caamaño Olivares de Granada que aparece en el numero 5 en la que Guillermo dice no conocer el comando "GO" al igual que la redacción

Bajo mi modesta experiencia quiero informar de lo siguiente "GO" dice una vez traducido "IR" y "TO" dice "A", tanto "GO" como "TO" son variables no utilizables puesto que las dos palabras forman la sentencia di bifurcación incondicional de un programa escrito en "BASIC" cuando decimos "GOTO 100" el programa bifurca a la línea cien, también podriamos decir "GO TO 100", traduciendo diriamos "IR A 100". Pruebe a utilizar la variable "GO" y "TO" vera cómo le dara error, también la sentencia separada tal como está escrita arriba vera como si le funciona.

TENEMOS LO QUE BUSCAS

PARA TUS HIJOS

- 1 VIC-20
- 1 LECTORA
- 1 MANUAL
- 2 JUEGOS
- 1 JUEGO AJEDREZ
- 1 CARTUCHO JUEGOS

TODO POR 61.890,o 5.900 PTS MES

PARA TU COMERCIO

- 1 COMMODORE-64
- 1 UNIDAD DISCO
- 1 IMPRESORA
- 1 TELEVISOR
- 1 PROG CONTAB
- GENERAL 1 BASE DE DATOS

TODO POR 349.950,-O 18.500 PTS MES

Y CUALOUIER OTRA SOLUCION PARA TI CON IDENTICAS FACILIDADES DE PAGO ADEMAS

CURSOS DE INICIACION ASESORAMIENTO INFORMATICO
CLUB DEL ORDENADOR



SERVICIO EN TODA ESPAÑA

COMPUTER CENTER

RBLA CATALUNYA 50 1 2º APART CORREOS 5455 08007 BARCELONA TELF (93) 215 60 70

... PREGUNTONES,

Actualmente trabajo con equipos IBM en la Empresa en la que presto mis servicios, también dispongo de un equipo CBM de propiedad motivo por el que estoy suscrito a la revista (en el MANUAL DE REFERENCIA COMMODORE hacen alusión aunque muy simple de lo refendo antenormente)

Próximamente me pondré en contacto nuevamente para asi ser un colaborador más de la revista, mientras, saludos cordiales

RICARDO ROCA

C/EXCORXADOR, 6, S/A 2º VILAFRANCA DEL PENEDES (BARCELONA)

DIVERSE



Me gustaría saber si exise alguna manera de utilizar el Joystick en un programa ya hecho (o sea un programa para utilizar el Joystick)

fambién quisiera intercambiar juegos con otros lectores de esta magnifica revista

LUIS ALFONSO SANZ Pº SAN ISIDRO, 5, 2º A VALLADOLID-12

Se puede utilizar. El ordenador lee la posición del joystick en las direcciones: 37137 y 37152 (VIC-20)

56320 y 56321 (C-64): Una para cada Port.

COMPATIBILIDAD DE CARTICHOS



Al cargar el VICPRINT y otros programas de cinta, si además del cartucho de 16k RAM tengo puesto el superexpander, el ordenador se

expander, el ordenador se queda bloqueado. Si quito el super-expander todo va bien

¿Es ésto normai? VICOLAS DOMINGUEZ

Normalmente los dos cartuchos son compatibles, pero determinados programas que interceptan el sistema operativo KERNEL, son incompatibles con el Super-expander que también lo intercepta. Esa es la causa de tus problemas, y como ves es completamente normal en determinados programas. La solución suele ser, dejar el super-expander en el armario y sacarlo sólo cuando lo necesites de verdad.

MASS OF DAROS



Me llamo Jorge Llambi González, tengo 13 años y poseo un Commodore 64 Paso a exponeros el problema que tenia

En el programa base de datos de la revista nº 3 pagina 60 al grabar algun dato no me funcionaba, cambie la linea 400 tal como sugeriais en la revista nº 4 en meteduras de pata, pero me seguia dando error en la linea 910

Cambié en dicha linea et primer print#] por print#1 y me dio error en la linea 920, en esta hice exactamente lo mismo; cambie el primer print#] por print#1 y funciono JORDI LLAMB

JORDÍ LLAMB APARTADO 102 BLANES (GERONA)

Seguramente olvidaste poner el "J=1" que aparece en la linea 400 del programa. Si la hubieses puesto habria funcionado a la perfección

CODIGOS DE COLORES



Amigos de Commodore World Hola¹ Me llamo Ricardo Poseo un COMMODORI 64 desde hace unos meses. N

desde entonces estoy suscrito a vuestra revista. la cual encuentro muy interesante

Os escribo, para subsanar un fallo del manual del Usuario del COMMODORI: 64 En el apendice F, en el que se relacionan los codigos CHRS, no aparecen los correspondien-

Service E)

tes a los colores accesibles a través de la tecla LOGO de COMMODORI

A continuación los relaciono-

CHR\$ (129) Naranja CHR\$ (149) Marron

CHR\$ (150) Rojo claro

CHR\$ (151) Gris uno

CHR\$ (152) Gris dos CHR\$ (153) Verde claro

CHR\$ (154) Azul ciaro

CHR\$ (155) Gris tres

Y nada más por ahora Hasta mi proxima colaboración, ¡ADIOS COMMODORIANOS! RIC 4RGO G 4RDES MOLL CHILLESTA J 5 12

ESPECGAS DE LEOBREGAT (BARCELONA)

SUGERENCIAS



Hemos recibido cartas sugiriendonos que adaptemos todos los programas publicados al otro ordenador, los del VIC-20 al C-64 y vice

versa, e incluso que saquemos un par de nume ros extra con todo lo publicado adaptado Como respuesta en general, los que sean

Como respuesta en general, los que sean colaboraciones podeis poneros en contacto con los autores, o con los clubs de ordenadores de vuestra localidad, seguramente a otro también le interese la idea y podáis compartir el trabajo entre varios, ya que nosotros no podemos emplear todo ese tiempo en adaptar programas que no son nuestros para los otros modelos de ordenadores.

En los programas que sean nuestros, casi siempre apareceran dos listados uno para el VIC-20 y otro para el C-64. Cuando no apare cen es porque el programa utiliza particularidades del sonido o gráficos del ordenador, y la adaptación sería muy difícil. También nos han sugerido que publiquemos programas para el Simons Basic en una sección exclusiva para él Creo que esto sería injusto, pues deberíamos tener una sección FORTH, otro PASCAL, otra para el VICTREE, otra para el Berdens Basic, etc. Como no caben todas en la reista, sacaremos programas para lenguajes respectivos, pero no en todos los números.

También piden articulos sobre perifericos, hard, etc Todo lo que nos mandan las casas distribuidoras lo sacamos en la sección Comentarios Commodore, ya iremos publi-

cando lo que nos envien

Algunos piden que les enviemos listas de todo lo que hay en el mercado, como no tenemos estas listas (sólo las que vamos publicando), a continuación van las señas completas de los que desean recibir listas de productos para que las casas se las envien: Miguel Bahr Cruz, C/Matola, Km 2, Villacruz, nº 2, Elche (Alicante); José Luis Ranz Yubero, C/Doctor Fleming, 13, 3º izda Guadalajara

Además de la sugerencia del cambio entre VIC y C-64. José Luis Ranz nos pedía programas de gráficos en tres dimensiones, y queria saber como adaptar "evaluación de colegios"

al datassette

Los programas de graficos están y seguiran saliendo, ya que hemos recibido una lluvia de colaboraciones sobre este tema

El programa evaluación de colegios, no se puede adaptar para luncionar con cassette, ya que utiliza ficheros de tipo de acceso directo, y el datassete solo puede manejar ficheros secuenciales

ELECTROAFICIÓN COMPUTER

C/ VILLARROEL, 104 - 08011 BARCELONA - TEL. 253 76 00 - 09

PRODUCTOS COMMODORE

Commodore-64 Disk Drive 1541 Cassette CN2 Monitor Color 1701 Impresora MPS-801 Commodore 64SX Portable VIC 21

SINCLAIR

Spectrum 48K Impresora Setkosha con interface Microdr ve Teclado DKTRONICS LAPIZ óptico Amplificador Sonido

SOFTWARE

Contabilidad Domestica Control de Stocks Mailing y Etiquetas Licheros Base de Datos Gran variedad de Juegos Programas Educativos

IMPRESORAS

Serkosha Star Epson NewPrint C Itoh Riteman

Interfaces

GAMA COMPLETA DE ACCESORIOS

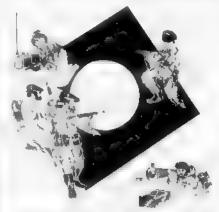
Jovsticks Sintertizadores de voz Cassettes Cuntas Discos Base de Datos Easy Script Monitores Interpod Cables Procesador de Textos Libros

ORDENADORES DE GESTION

Pal Computer Commodore Apple continuación del listado del volcado hexadecimal

0868 56 49 43 20 42 59 45 4E 59 2870 48 2E 20 210 9E 0878 89 86 1919 32 22 00 93 AC. 34 34 0840 35 36 28 29 AA. 0.2 28 34 33 29 AA **BBBB A5** 20 99 99 85 34 38 00 0630 A5 22 A5 SE. 85 23 37 65 9898 85 25 ĤИ 00 AS BHHB 24 H5 30 Dià 02 CE 06 22 **B**1 22 23 йьнн. A5 22 82 66 Dи 22 LØ 30 OB BU 8696 23 **C6** 22 B1 22 FØ. 21 85 23 26 85 22 Dist 62 Úб 6 BRCB. 8A H5 B1 22 18 65 24 0808 **A5** 37 DO 02 65 25 48 BURG 26 91 Ľ6 37 66 37 BA **BUDS** 06 38 06 **H**5 37 Dø 02 38 06 BUED 48 37 91 37 18 90 B6 **C9** 68 Balas BF A5 37 85 33 **A5** BEF 0 DØ ED 08F0 A2 38 65 34 60 37 ЙW HF 93 FF 00 A9 98 00 BD BUEB 00 00 00 CB 07 7E 40 FØ 99 **B908** 86 **E**3 HØ ØR 0910 EB 110 F4 4C FØ 32 E9 30 36 BB 09 20 9918 FD FF 00 88 0928 E9 DØ RD 68 99 SE FE EF 20 0928 SD FC FF 01 FF BH BD Ø1 00 MPRN HC BD 07 02 01 09 **BEE** 0.9 BC DE **64KP** H4 FU 12 AC. FF FF 60 AE FF 00 AD FC FF 00 48 FF 0 - 48 73 FD 28 60 40 HD EE Ois BCEB FF 0958 00 00 AD FC. 00 48 914 JH. **A2** 80 OB 46 ØB **A4 7**A 09b0 FD 98 99 99 02 38 89 BUED 80 F0 E6 09 13 9979 MO FW 13 E BL 98 FF 00 10 EH 8778 MB E4 F8 FF 00 D0 Ch HD 916 0900 DØ. EB 84 26 H5 ØB 6999 ES CB E3 FF 00 HH BD 48 HI BERB ØH. 48 20 7B FC 88 FF ØB. 693d E2 73 00 00 68 68 20 71 **BHY** 4C FL 48 68 89 40 УF **A4 BREN** ЙИ D0 02 **A5** 20 66 2E 07**0**0 HØ. 20 2D 7B 21 88 DØ F5 **PI**5 85 0988 **"**6 FB FC 40 FL A5 2E 85 **0**900 20 29 FF 49 24 ВD 90 de 80t0 30 83 01 03 H2 HØ HH UPLU 20 BD 60 BIED FF A9 ØE. Ald **A2** BR FF 20 CB H9 09EU ИĤ 210 FF A9 60 213 96 83c0 98 20 84 **A9** 88 ଧାଧ 85 90 ĤØ 03 FF A5 ac 30 03 20 FF 85 FD R9F8 FF **ОНЙЙ FI4** 90 DØ 39 20 H5 **d**5 FE **A4** 90 DØ. 30 HC 34 83 BUENS. DO E5 A6 FD н5 FE 68 20 dH1d 20 CD PD 78 FF MB н5 **BA18** 210 19 09 ÜU FF 90 DO 00 MHIN Нь. FØ EE 20 De 213 E1 0H28 ME 0830 F0 OD 20 66 F.F. 80 4C 42 FD DO. 20 FF ЙИ HØ 28 MH.38 BB A9 ØE 20 40 De 03 инай. 20 **Ø848** CD. FF 40 86 E3 20 4C 86 E3 85 28 85 FB 0A50 90 FC 85 20 ED FE 90 20 ชะเหย H5 00 **R9** विध विध Ĥ6 BHEB 20 BB FE F H 84 FC 20 05 86 2Dйнья EH EH EH ER Eñ 0A70 **H4** 2E ER 33 H5 ER OH/H EA EA EH 20 210 59 **R6** 40 BB FD ИÜ 210 **BRHB** FE 99 A5 21 FE 00 20 DΘ **Ø**688 85 85 FL FB ØH9Ø FB Н5 20 НŶ 20 Ĥ4 2E 20 Da FF 211 ØH9B 86 FF 40 20 92 FE 60 40 24 **ØHHØ** 0AH8 86 E3 28 39 FF 20 EF 90 UHB0 FE 00 20 73 00 00 CY

0HB8 D0 2D 9D 3C 03 E8 20 73 8HC8 84 88 C9 49 D0 20 30 **ศิลิติ** MA MM EM 1 D 4D Из Ea OHLO 90 20 23 80 80 FØ 14 36 ĘΘ 73 80 ØÙ MHL8 20 9D 30 03 20 4F ий BARB B Lied 71 DHES 69 44 FF 80 40. 48 HE BHEB 20 29 FF 00 4C 66 EЭ 20 E2 4C 41 410 MAFA F£. Ha 94 29 FF 0600 213 30 02 210 BU did E7 Dw **E**3 A5 **BUd** FØ LS 22 FB 65 н5 7B 85 73 20 0510 FØ 09 Dia. **BB18** 00 610 RE 22 0320 20 73 ØÜ ЙÛ 09 54 Ūΰ CB 09 4F Lü [4 йи 08.8 20 73 ий EF FE **0B30** 20 UÜ 80 10 BIN **6878 A9** 3D 9L30 03 E8 **H**5 FB FL 85 85 78 **H5** 2 B 20 FF OR40 0948 FE Ød н9 ЙИ Йú 311 JĹ. 13.3 0350 40 71 FE 66 20 3F Nin EF FE HJ. UÜ 36 ØR58 214 Med ರಚ FF 4F ИÑ Del йвьй JC. 63 20 29 FE 40 71 00 20 EL FE Ud BEER **0870 A9** 86 U8 9D 30 03 فائح 23 FF FF 14% 3B'8 20 20 18 HÌ 1461 20 ивыи 40 ĤЭ 30 63 FR 46 Hai F5 FF Jbas FF C8 96 46 HE a li 92 FE ЙЙ FU 03 20 94 BB30 E3 40 4L Ĥ9 MR 85 RH BBJB 86 FF **ØBAÐ** 20 **B4** FF H9 ьF 20 96 BAAS AU. ØØ 00 20 **A5** FF 99 30 F5 **GPB9** 69 Ca 69 ab bu HJ Idid HB FF ObBo 00 99 30 03 20 HE DЭ 0500 BH 8H ЙĴ 30 باك **613** Det 15 **OBCS** 06 _Û L9 30 03 80 لائم FF 75 REPAR 24 FF 46 20 JĽ. 20 HÜH 63 -0 1E **BBDd** EF JU ĤĤ 30 ULLU الأ HB 50 to 4å D ŞΈ EJ 60 8336 29 FF UÜ 8A H2 30 HU 43 FE 89 Ян 80 05F0 28 БL 416 HØ FF 05F8 ÜØ. 84 90 40 BA H2 MB 20 79 20 60 0000 99 36 **B**7 CS 2A 15 22 Ind 1 h En DМ OC 48 0010 0a E6 7B RU 80 136 B1 **ZH** F0 90 0018 F0 ĞН 69 22 Øb FU 87 01 F4 03 Ea D0 EA WC20 €8 4C HF 0.28 60 58 do HJ Ud 0030 20 B1 FF A9 ьF 40 33 H# 00 40 C3 FF 69 41 0038 21.



20 1A FF 56 49 66 6H 0040 R9 FF 0048 20 R8 FF 4L HE UC50 A₃ 52 20 HJ 53 A2 20 ES AY 38 90 30 ชบ5ช 88 9L 03 £ы 80 ИИ 0,00 30 **U** 3 60 Ha EF. 7DFF 00 FO ÖÞ 16 Miloo F5 ĆF 53 0.9 0C70 EB Lu 20 0078 ØB. 20 75 FF 00 28 60 26 FO AL 80 £4 FF 69 69 20 FF FØ FB H9 JŪ E4 611 **ರ**್ಷ ಭಚಿ 20 40 D2 EE aD 45 ZC. 89 28 UL 36 53 47 41 53 45 54 20 0098 53 53 28 JE 55 52 4F 4F 40 BUHB 4E 23 20 00 ЫÜ 46 **BUHB BA** 09 20 **80 680** 00 MA RØ MI 4E GC 18 41 50 54 45 64 43 41 45 07 4F 40 00.00 54 41 43 45 43 **D4** 44 4C 4F 41 40 BULB 53 5ь 65 45 Del. Ĺ4 44 41 48 49 O Da 41 44.4 45 Le 4F 49 DA 4F 41 49 40 46 Úь 52 MILER Li ŭ5 53 41 45 52 OCES. 45 4E 43 41 54 43 45 4F 6.4 WLF0 L d 55 53 54 41 54 иСЕв 1.7 FU Få FI 410 FF 6114 6C OU UD **BUJS** 66 72 FD שא שש FIL Mid f at FD FL FL Ud 03 OI Int Hild FE 55 OU UB ÜU FE UB FE WI-1d FF MB ALISA AR FR FE ИÙ 66 UB 00 00 00 00 00 NO NA UB ЫU

PROGRAMADORES

Editamos y comercializamos programas para ordenadores personales (CBM64, SPECTRUM, etc.)

USUARIOS

Necesitamos colaboradores para traducción de programas y manuales. Imprescindible disponer de equipo.

CASA DE SOFTWARE, S. A. Aragón, 272, 8º 6º 08007 BARCELONA
Teléfono: (93) 215 69 52







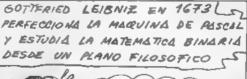












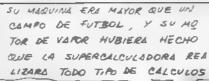


O: NO EXISTENCIA

JACQUARD EN 1801 CONSTRUYE ON TELAR QUE EFECTUA DIBNIOS EN LAS TELAS GRACIAS A UNA TARJETA QUE LLEVABA UNOS AGUJEROS EN SITIOS ESPECIALES









PERO NUNCA PUDO TERMINARIA EL IN
VENTO NO FUNCIONO, NO SE HABIA
INVENTADO AVN LA ELECTRICIDAD
I PERO SUS IDEAS SI QUE FUERON UTI
LIZADAS MAS ADELANTE

LESTUVISTE MUY
CERRA CHARLY



Y NO DIVIDENOS A LADY ADA LOVELACE
HIJA DEL POETA LORD BYRON .. FVE
NADA MAS M NADA MENOS QUE LA
PRIMERA PROGRAMADORA , APOYANDO EL
LA INFORMATICA

WILLIAM BURROUGHS (1857-1898)
ABVRRIDO POR SU TRABAJO DE CON
TABLE CONSTRUYE UNA MARVINA
DE CALCULO CON LISTADO INCOR
PORADO ... I LA PRIMERA COLCULADORA DE NESA!

UN GRAVE PROBLEMA; EL BENSO

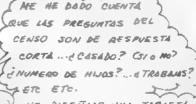
(ADA DONTEO TARDABA 2 AÑOS Y
MEDIO Y CLBRO, NO ERA FIABLE

CORRIJE)



CON 63 000 000 DE PERSONAS Y LA
CONTINUA INMIGRACION ERA IMPOSIBIE
COMODERIO CON PRECISION, PERO
APARECIO EN ESCENA HOLLERITZ

NE ME DADO CUENTA



QUE HACE FUNCIONAR MARLUMAS Y BLA BU

HE DISENADO UNA TARJETA)

DONDE EN LUGAR DE ESCRIBIR

NAPA, HACEMOS UN AGUJERO

EN LA CONTESTÁCION CORRECTÁ

LAS TARIETAS SE RELLEMA

BAN CON FACILIDAD Y RA

PIDEZ SE MARCABA ÉL

SITIO DE CADA RESPUESTA

Y YA ESTABA

LAS LEEMOS?

























VIDEO CASINO

Carrera de caballos

Gabe GARGIULO (RUN EEUU)

fraducido por Valerie SHANKS y adaptado por Alvaro IBAÑEZ

Guarda tus prismáticos y vente al hipódromo (con este programa y tu VIC-20 lo podrás hacer sin perder dinero).

PARA TU VIC-20 - NO AMPLIADO

Este programa de carreras para el VIC-20 no ampliado utiliza el símbolo que se parece un poco a un caballo, o un perro o un pollo, con un poco (a decir verdad, con cantidades industriales) de imaginación. Mi versión sólo funciona en el VIC, ya que utiliza pokes

l'ú empiezas a jugar con \$500, y puedes apostar hasta esa cantidad. Tú escoges un caballo, de 1 a 5, para ganar. El programa elige el ganador al azar. Si tú has apostado por el ganador, ganas cuatro veces la cantidad apostada, que se añade a la cantidad que tienes. Si pierde tu caballo, tú pierdes la cantidad apostada. Puedes jugar hasta que pierdas todo el dinero que tentas. (Esto es inevitable.)

Es fácil teclear este listado, el juego resulta divertido, y sobre todo, no te cuesta nada. El listado demuestra un buen estilo de programación, que si lo sigues, da buenos resultados que son fáciles de comprender y modificar Empieza con un REM (comentario) que demuestra el nombre y finalidad del programa. Ahora lista las variables y su explicación. A continuación, presenta la lógica principal del programa, que contiene sus decisiones más importantes. Coloca las subrutinas, que se utilizan con las sentencias GOSUB, después de la lógica principal del programa.



Tabla 1. Descripcion del programa principal

Nº DE LINEA

15 FIJA RI (CANTIDAD DE DINERO EN MANO) EN 500

10 EDA RITCANTIDAD DE DINERO EN MANO) EN 500
20 BORRA LA PANTALLA. LLAMA LA SUBRUTINA 1200 PARA
COLOCAR UNA FILA DE GUIONES ATRAVESANDO LA PANTALLA DESCRIBE LAS VARIABLES M... LA.
22 FIJA M. (POSICION DE MEMORIA) EN 7680, LA DIRECCION
INICIAL DE LA PANTA
LLA FIJA LA (Incess transversales) EN 22. POKE 36879- FIJA EL
COLOR VEL FONDO.

40 LLAMA LA SUBRUTINA 1200 PARA QUE COLOQUE UNA LINEA DE GLIONES EN LA PANTALLA. 80 PONE EL TITLE DE LE SMELLA SUBRUTINA 1200 PARA QUE

COLOQUE UNA FILA DE GUIONES EN LA PANTALLA

- 85 DESPUES DE UNA CORTA ESPERA ARRANCA L. 1980. GRAMA
- 86 FIJA LA POSICION INICIAL DE LOS CABALLOS

- 90 HO DA LAS INSTRUCCIONES 140 LLAMA LA SUBRUTINA 1200 PARA QUE COLOQUE UNA FILA DE GUIONES EN LA PANTALLA 270 APUNTA EL NUMERO DEL CABALLO APOSTADO 300 LE ANY LA SUBRETINA 200 PARA QUE COLOQUE
- DE GLIONES EN LA PANTALLA
- 310 SOLICITA LA APLESTA
 320 STLA APUESTA ES MENOR QUE O IGUAL A LA CANTIDAD EN
 MANO, SE AVANZA A LA 400
 330 (1 ST NO LO ES) INDICA LA CANTIDAD QUE QUEDA PARA
 APOSTAR

350 SOLICITA UNA APLESTA DE NUEVO (310) 400 BORRA LA PANTALLA Y LLAMA LA SUBRUTINA 2000 PARA QUE PONGA LOS CABALLOS 410 ELAMA LA SUBRUTINA 4000 PARA QUE PONGA LA PUERTA DE SALIDA 600 FLIGE UN NUMERO AL AZAR ENTRE LY 5
620 ELAMA LA SUBRUTINA 1000 PARA QUE ESE SUME A UN CONTADOR QUE CORRESPONDE AL CABALLO GANADOR ELAMA LA SUBRUTINA 2000 PARA QUE PONGA LOS CABALLOS UN LA PANTALLA (SELECONTADOR DE UN CABALLO HA SIDO INCREMENTADO UNIL AVANZA) SE SUMA I AL CONTADOR QUE CORRESPONDE AL CABALLO GANADOR 640 SI EL CABALLO QUE ACABA DE DESPLAZARSE NO SE ENCUENTRA CERCA DEL LADO DERECHO DE LA PANTA-LLA, EL PROGRAMA VA A LA 600 PARA QUE SE MUEVA OTRO CARALLO 650 SUHA HABIDO GANADOR, EL PROGRAMA ESPERA UN POCO 660 ANUNCIA AL GANADOR 665 COLOCA UNA FILA DE GUIONES POR LA PANTALLA 670 SI EL CABALLO ELEGIDO ES EL GANADOR, SE SUMAN LAS GANANCIAS A LA CANTIDAD EN MANO: EL PROGRAMA AVANZA A LA 750 680 SI EL CABALLO ELEGIDO NO ES EL GANADOR, EL PRO-GRAMA HACE UNA PAUSA AQUI APARECE EN PANTALLA EL MENSAJE "TU PIERDES" RESTA LA CANTIDAD DE LA APUESTA DE LA CANTIDAD DE DINERO EN MANO 685 INDICA LA CANTIDAD DE DINERO QUE QUEDA 687 SI NO QUEDA DINERO, APARECE EL MENSAJE "ESTAS BROKE" EL PROGRAMA TERMINA 690 TE PREGENTA SEQUIERES JEGAR OTRA VEZ 69 RECIBE RESPUESTA 700 SELA RESPUESTA LS ST. E PARA EMPEZAR DE NUEVO , 11 PROGRAMA VUELVE A LA 20 710 SI LA RISPIESTA NO ES S' ACABA AQUI RESTAURA EL COLOR DE PANIALLA Y 11. FONDO FINALIZA EL PRO-GRAMA 766 PONE LA CANTIDAD DE DINERO EN MANO 770 EL PROGRAMA VUELVE A LA 690 PARA PREGUNTAR SI QUIERES JUGAR DE NUEVO.

Tabla 2. Descripciones de las subrutinas

990 FINALIZA

VARIABLES

SUBRUTINAS
1000 SUMA LA X1, X2, X3, X4 O X5, SEGUN EL NUMERO QUE HAYA
SALIDO AL AZAR
1200 COLOCA UNA FILA DE ASTERISCOS POR LA PANTALLA
2000 HACE AVANZAR LOS CABALLOS CUYO NUMERO HA
SALIDO DEJA LOS OTROS CABALLOS EN SU SITIO.
1000 PRODUCE EL RUIDO DE LA PISTOLA DE SALIDA Y EL
GALOPE DE LOS CABALLOS

Tabla 3. Definiciones de las variables

MI POSICION EN LA MEMORIA DE UN CABALLO LINEAS HORIZONTALES EN PANTALLA, 22 EN EL VIC POSICION DEL CABALLO I POSICION DEL CABALLO 2 ΧI POSICION DEL CABALLO 3 POSICION DEL CABALLO 4 POSICION DEL CABALLO 5 Ri CANTIDAD DE DINFRO EN MANO VARIABLE DE INDICE NUMERO ELEGIDO AL AZAR CABALLO POR EL QUE SE HA APOSTADO CANTIDAD APOSTADA "ARRAY" EMPLEADO PARA PODER LOCALIZAR A CADA CABALLO, B(1) ES PARA EL CABALLO #1, B(2) ES PARA EL 800 CABALLO #2, ETC RESPUESTA S/N 7.5 VARIABLE OF INDICE COORDENADA "Y" EN PANTALLA (EL NUMERO DE LINFAS A PARTIR DE LA PARTE SUPERIOR) VARIABLE DE INDICE .13 VARIABLE DE INDICI J2 VARIABLE DE INDICE VARIABLE DE INDICE VARIABLE DE INDICE

VIC-20

10 REM CHRRERMS 15 RI=500 REM RI=DINERO QUE QUEDM 20 PRINT"[CLR][WHT]" REM ML=LOCALIZA CION DE LA PANTALLA

22 ML=7680 LH=22 PUKE3687978 30 B=0 R=0 FUR1=1105 HCI>=0 Bc1>=0 N EXII 4M GUSUB12MM 80 PRINT"BlenvenibulsPulhisPulEHs195Pul CHRRENHS125PC 1" GUSUB12WM 85 FORT=1T0500 NEXT REM AUUI COMIENZ A EL JUEGO 36 X1=1 X2=1-X3=1 X4=1-X5=1 REM CULU CH A LOS CHAHLLOS 90 PRINT"TIENESCSPCJ\$", R1, "PHRH" PKI NT "APUSTHR. TWO PRINT"ENSESPECIMPUESTHSESPECIESTHN ESSPUJ4 1" - PRINT 110 PRINT"ELIGE(SPCJUN(SPCJCHBALLUISPC) (1-5)140 GUSUBIZUU 270 INPUT*[SPC]*.H 288 IFHOUNNIHCE THENSIE 290 PRINT"LOSISPUJCHBHLLUSISPUJESTAN [4SPC]NUMERHDUStSPC]DEE[SPC]11SPC]HL ESPECIS. ESPECIANTRUDUCEESPECIELESPECINUM ERU 390 GUSUB1200 GUTU270 310 PRINT"CUHNTOESPOJAPUESTHS": INPU TR B=HB5(B) 320 IFBORT OF BERTTHENAUG 330 PRINT"TIENES[SPC]\$",R1,",[SPC]", 350 GUTUJIU 400 PRINT"[CER]", GUSUBZUBU-REM CULU CA LOS CABALLOS EN SU SITIO 410 GOSUB4000 REM LINEA DE SHLIDH 6UU R=INT(5#KND(1)+1) 620 GOSUBINNO GUSUBZNOO 630 B(R)=B(R)+1 640 IFB(R)<(LH-Z)THENBUU 650 PRINT"[SPC]" FURI=1101000 NEXTI 660 PRINT"[CLR][2CRSRDJEL[SPU]GHNHDO FISPCIESISPCIELLOSPCINUMERO", R 665 FURIZ=1TOLH PRINT"=" , NEXTI2 670 IFH=RTHENR1=R1+(4#8) (GUT0750 680 PRINT"HHS[SPC]PERDIDO, " - R1=R1-B 685 PRINT"[CRSRD]TIENES[SPC]\$",R1 687 [FR1C=0]HENPRINT"[CRSRD]ESTHS[SPC] HRRUINHUO. VUELVEOTROISPOIDIH" END 698 Z##"" PRINT"[CRSRDJUTRH[SPC]CHRK ERR(S/N)" 691 INPUT2\$ 700 IFZ\$="5"1HEN20 710 POKE36879,27 PRINT"[CLR][BLU]", END 750 PRINT"[CRSRD]HHS[5PC]GHNHD0[SPC] \$",4#B 760 PRINT"[CRSRD]HHORH[SPC]TIENES[SPC] #">R1 770 GOTU698 990 END 1000 REM SUMP AL CONTADOR APRUPIADO 1090 IFR=1THENX1=X1+1 1092 IFR=2THENX2=XZ+1 1093 IFR=3THENX3=X3+1 1894 IFR=41HENX4=x4+1 1095 IFR=5THENXS=X5+1 1100 RETURN 1200 PRINT FORJ=1TOLA:PRINT"-"; NEXT J PRINT RETURN 2000 REM COLOCA LOS CABALLOS EN LA P ANTALLA 2010 Y=1 PRINT"[CLR]" 2020 POKEML+X1+22#Y,94 2021 POKEML+X1-1+22#Y,48+1 2022 POKEML+X1-2+22#Y,32 2030 Y=5 2040 POKEML+X2+22WY, 94 2041 POKEML+X2-1+22#Y,48+2 2042 POKEML+X2-2+22#Y,32

2050 Y=9

2060 PUREME+X3+22#Y,94 2061 POKEML+X3-1+22#Y, 48+3 2862 POKEML+X3-2+22#Y, 32 2070 Y=13 2080 POKEML+X4+22#Y,94 2081 PUKEML+X4~1+22#Y,48+4 2882 POKEML+X4-2+22#Y,32 2899 Val 7 2100 FUKEML+X5+22#7,94 2101 POKEML+X5-1+22#Y,48+5 2102 PUKEML+X5-2+22#Y,32 2130 RETURN

3000 REM SONIDO 3010 POKE36878,15 POKE36874,200 3020 FORJ3=1T05 NEXTJ3-PGKE36074.0 3030 POKE36878.0 RETURN 4000 REM SONIDO PARA COMENZAR EL JUE GO Y DISPHRO 4010 FORY=0T022 PUKEML+3+22#Y,115 NE XTY 4020 FORJ2=1701011 NEXTJ2 4040 PUKE36877,220 FURL=1870057EP-1 4050 PUKE36878, L. FORM=1T070 NEXTM-NE XTL:PUKE36877.0 POKE36878.0 4100 RETURN

ESTE ES PARA TU C-64

A prueba de reflejos

Soltan SZEPESI (RUN EEUU) Iraducido por Valerie SHANKS y adaptado por Diego ROMERO

Unas secuencias de color y sonido ponen a prueba tu memoria y velocidad de reflejos en este divertido programa.

El programa "Repetir la Secuencia" te permite jugar a tres cosas distintas que te hacen ejercer y le ayudan a mejorar lu memoria tanto visual como auditiva. Otro juego de este programa resulta útil para comprobar tus reflejos. El programa (Listado I) está escrito en Basic para el C-64, pero puede ser modificado para ser usado en otros equipos.

La idea detrás de estos juegos es la misma que la de "Simón". Sin embargo, al utilizar el ordenador y su pantalla, se logra una mayor comunicación entre la máquina y el

Cuatro cuadrados de diferentes colores aparecen en cuatro sitios distintos en pantalla. Salen unos colores intermitentes aleatorios, casa uno acompañado por un sonido especial. Tú lo tienes que repetir pulsando la misma tecla de color (sin usar la tecla de control). Los colores utilizados son púrpura, verde, azul y amarillo: por lo tanto, las teclas 5, 6, 7 y 8 tienen que ser pulsadas. Si repites los colores y los sonidos correctamente, el juego continúa. A continuación se presenta la descripción de los cuatro juegos.

Juego 1. Crear la secuencia.

Después de haber recibido la primera señal del C-64, tienes que añadir otra señal. A continuación, tienes que repetir la secuencia de las señales anteriores y añadir otra. Continúas de esta forma hasta completar un número determinado de pasos (ver la Tabla 1), y el C-64 te saludará con las primeras ocho notas de la Quinta Sinfonia de Beethoven. Si no lo has hecho con suficiente rapidez, o si te has equivocado al repetir la secuencia, el ordenador te toca una parte ruidosa de la Quinta Sinfonia y el juego termina

Antes de empezar cada juego, puedes elegir uno de los cuatro niveles de dificultad pulsando una de las teclas de funciones programables (F1, F3, F5 o F7). La Tabla 1 lista los distintos parámetros definidos por estas teclas.

Cuando se pulsa la tecla de función F3 (después de que el C-64 lo solicite al iniciarse el juego), se tienen que repetir



correctamente ocho secuencias para poder terminar el juego. El tiempo transcurrido entre cada señal será muy corto: tienes que pulsar el color correcto a los tres segundos de haber pulsado el color anterior.

Juego 2. Repetir la secuencia.

El C-64 empieza por dar una señal. Al repetir tú la señal correctamente, el ordenador repite la señal anterior y añade una señal nueva. Tienes que volver a repetir esta secuencia de nuevo. En los pasos siguientes, el 64 repite la secuencia anterior y añade una señal nueva hasta que se complete la serie de acuerdo con el número de secuencias elegido.

Juego 3. Cambiar la secuenca.

Este juego se parece mucho al Juego 2. La única diferencia es que el ordenador no repite la secuencia dada anteriormente. sino que empieza una nueva secuencia con una señal adicional.

Juego 4. Intermitencias Simnles

En este juego, el C-64 sólo da una señal a la vez y tú dispones de un segundo para repetirla.

La Tabla 2 presenta los números de las secuencias y el contenido de cada sección del programa. La lista de las variables principales se presenta en la Tabla 3.

Las variables N1, T1 y T2 son fijadas por las cuatro teclas programables en las sentencias 60 a 75. Pueden ser modificadas si se cambian los números en estas sentencias. El tiempo transcurrido entre cada señal (T1) es totalmente un valor relativo. Este se añade al tiempo que se tarda en representar el cuadrado de color. T2 está en segundos. En el Juego 4, T2 se vuelve a definir en la sentencia 355. Dado que el programa sale de la subrutina "REPITE LA SECUENCIA" (las secuencias 450 a 485) en unas circunstancias determinadas sin utilizar el comando "Return", el "stack" puede llenarse (después de unas 13 partidas en el mismo número de juego) y podría presentarse un mensaje de error "OUT OF DATA". Para evitar este desastre, se introdujeron tres comandos adicionales "Return" con la bandera Q. La subrutina "FOR STACK CLEARING" (las sentencias 700 a 710) proporciona la sentencia correcta "GOTO" (de salto).

Una operación semejante de llenar los "stacks" puede realizarse cuando queda un bucle For... Next antes de terminarlo. La primera parte de la sentencia 515 soluciona este problema.

El programa requiere 3568 bytes de memoria; se necesitan 400 bytes adicionales para ejecutar el programa.

	Tabla 1. Los nive destro de un jue	_	
TECLA DE FUNCIONES	NUMERO DE SECUENCIAS	TIEMPO ENTRE SEÑALES	TIEMPO PERMITIDO AL JUGADOR
FI	4	100	5 SEGUNDOS
F3		50	3 SEGUNDOS
F5	16	01	2 SEGUNDOS
F7	32	10	1 SEGUNDO

Tabla 2. Lista de secciones principales del programa

SENTENCIA	Nδ	TEMA	Y	COMENTARIOS

5-10 TITLE O Y AUTOR

15-95 INICIALIZACION ELEGIR NUMERO DE JUEGO Y NIVEL DE DIFICULTAD

200-230 PROGRAMA PRINCIPAL DEL JUEGO I

250-275 PROGRAMA PRINCIPAL DEL JUEGO 2 300-310 PROGRAMA PRINCIPAL DEL JUEGO 3

350-360 PROGRAMA PRINCIPAL DEL JUEGO 4

400-435 SUBRUTINA DE COLOR Y SONIDO

450-485 SUBRUTINA PARA REPETIR LA SECUENCIA

500-520 MENSAJES DE ERROR 525-560 PROGRAMA DE MUSICA

570-580 FINAL CORRECTO, QUINTA SINFONIA, (DATOS DE LAS LINEAS 555-560)

600-640 REPETICION DE LA ULTIMA SECUENCIA CORRECTA³

650-680 OUIFRES CONTINUAR"

700-710 BORRADO DE LOS "STACKS"

Tabla 3. Lista de las variables principales

VARIABLE COMENTARIOS

11 CURSOR ABAJO + CODIGO PURPURA

MARCA LAS MANCHAS DE COLOR CS POSICION DE MEMORIA DE COLOR

NUMERO DE PARTIDAS DENTRO DEL MISMO TIPO DE HIEGO

K(N) POSICION DE LA MANCHA DE COLOR EN LA INTERMITEN-CIA N DE LA SECUENCIA

MAXIMO NUMERO DE INTERMITENCIAS EN UNA SECUEN-CIA

NUMERO DE NOTAS EN UNA SECUENCIA DETERMINADA NM

NUMERO DE INTERMITENCIAS DEL JUEGO 4 n

BANDERA AL SALIR DE UNA SUBRUTINA POSICION DE MEMORIA DE PANTALLA

I.S **DURACION DE LA MUSICA**

NUMERO DE DIRECCION DEL SONIDO

TIEMPO TRANSCURRIDO ENTRE SEÑALES (VER EXPLICA-T1

RETRASO DE TIEMPO PERMITIDO, EN SEGUNDOS, AL REPETIR LA SEÑAL

ESTADO DEL RELOJ, AL MEDIR RETRASO DE TILMPO T2 CODIGO DEL TONO DE LAS NOTAS MUSICALES TN

TT NUMERO TOTAL DE PUNTOS DENTRO DEL MISMO TIPO DE DUFGO

NUMERO DE JUEGO

C-64

5 REM

10 REM

14 REM##INICIALIZACION############## ...

15 POKE53281,0.DIMK(32).B=0.TT=0.PRI NT"[WHT]

20 PRINT"[CLR][2CRSRD]##[SPC]MODALID ADESTSPCJDETSPCJJUEGO 155SPCJTCRSRD1 [5SPC]1.SECUENCIA[SPC]PROPIA"

25 PRINT"[CRSRD][4SPC]2.REPETICION[SPC] DELISPO I SECUENCIA:

30 PRINT"[CRSRD][48PC]3. SECUENCIALSPC] CAMBIANTET14SPC1CCRSRD1[5SPC14.DESTE LLOS(SPC) SUELTOS"

35 PRINT"[2CRSRD]##[SPC]PULSA[SPC]EL [SPC]NUMERO

40 GETAS IFAS ""THEN40

45 X=VAL(R#):PRINTX-IFXC10RX>4THENRU

50 PRINT"(2CRSRD)**ISPC]NIVEL[SPC]DE [SPC]DIFICULTAD(SPC)(CRECIENTE)[6SPC] [CRSRD][2SPC]PULSA[SPC]F1,[SPC]F3,F5 [SPC]0[SPC]F7"

55 GETA# IFA##*THEN55

60 IFA#=CHR#(133)THENT1=100 N1=4-T2=

65 IFR#=CHR#(134)THENT1=50-N1=8'T2=3

```
70 IFR#=CHR#(135)THENT1=10-N1=16-T2=
75 IFA$=CHR$(136)THENT1=1:N1=32:T2=1
77 1FT2=0THENRUN
80 B#="[10CRSRD][PUR]" - C#="[2SPC]PUR
PURRESSPECIORNIVERDECSSPECIEBLUIAZULESSPECI
EYELJAMHRILLÜ"
85 V=54296 - WA=54276 - A=54277 : SR=54278
 HF=54273 - LF=54272
98 PH=54275 PL=54274 S=1227:00=55499
95 NM=1 G=G+1:Q=0
100 PRINT"[CLR]"B$C$ W*RND(-TI)
105 ONXGOTO208,250,300,350
199 REMARSECUENCIAS PROPIASARARARARA
245
200 PRINT"[CRSRD]EL[SPC]C-64[SPC]DA[SPC]
ELESPOJPRIMERESPOJCOLOR" N=1 K(1)=I
NT(RND(1)#4) - GOSUB466
205 GOSUB450 IFQKOOTHEN760
210 NM=NM+1.N=NM PRINT"[CRSRD]AGREGA
[SPC]OTRO[SPC]CGLOR": T3=T1
215 GETAS : IFAS=" "ANDTI-T3CT2#60THEN2
15
220 IFTI-T3>T2#50THEN500
225 K(N)=VAL(R$)-5:60SUB460-60SUB450
 IFGCOTHEN700
230 BOTO210
249 REMARKEPETICION DE SECUENCIHSHAN
自由
250 PRINT"[CRSRD]ELTSPC]C-64TSPC]DATSPC]
LACSPOISECUENCIA" N=HM K(N>=INT(RND)
19#4):GOSUB400
255 GOSUB450 IFQC/0THEN700
260 FORI=1TUBBO NEXT
265 FORN=ITONM PRINT"[CRSRD]ELISPOJC
-6415PCIDATSPCILATSPCISECUENCIA" GOS
UB400 NEXTH
270 NM=NM+1 GOT0250
275 FOR I=1T0800 NEXT
299 REM##SECUENCIAS CAMBIANTES######
建油油
300 FORM=ITONM PRINT"[CRSRD]ELCSPC]C
-64(SPC]DH(SPC]LH(SPC]SELUENCIA" K(N
)#INT(RNJ(1)#4)
 305 GOSUB460 NEXTN-GOSUB450 FF0€>0TH
EN7UU
310 FORI=1T0800 NEXTI NM=NM+1 GOT030
 349 REM##DESTELLOS SUELTOS#########
 抽油油
 350 P=1
 355 T2+1 N=1-K(N)=INT(RND(1)#4)-GOSU
 BADD GOSUBASH IFQCOOTHEN700
 360 FOR!=1T0400 NEXTI: NM=1 N=1 P=P+1
 · 00T0355
 399 REMARDESTELLOS COLOR AND SOUNDAR
 由海南省
400 POKEY, 15 POKER, 0 PUKESR, 240 POKE
HF, 20+K(N) $5 POKELF, 10 POKEWA, 17
405 PRINT"[HOM]"B#+"[4CRSKD]PHSOLSPU]
 NUMERO: " :: IFX=4THENPRINTP - GOTO415
 416 PRINTN
 415 FORJ=8T03 FORL=CO+J#40+8#K(N)TQL
 +4-POKEL/KKN>+4 NEXTL/J
 420 FORJ=0T03:FORI=S+J#40+8#K(N)T0[+
 4 POKET, 160 NEXTI, J
 425 FORJ=0TOT1 NEXTJ : PUKEWA, 16
 43M PRINT"[CLR]"B$C$
 435 RETURN
 449 REM##REPITE LA SECUENCIA########
 自動車
 450 FORM=ITONM PRINT"[CRSkD]REPITE(SPC]
 LAISPOISECUENCIA" - T3=TI
 455 GETAS: IFAS=""ANDTI-T3CT2#60THEN4
 460 IFTI-T3)T2#55THENQ=1 - RETURN
```

```
465 Y#VAL(A$):IFY-5<>k(N)THENQ#2-RET
URN
470 IFY-5=K(N)THENGOSUB408 NEXTN
475 IFX=4THENN=P+1
480 IFN=N1+1THENG=3-RETURN
485 RETURN
499 REM MENSAJES DE ERROR*********
500 PRINT"[CLR][20KSRD]#[SPC]PERDIST
E-NOISPOJFUISTEISPOJBASTANTEISPOJRAP
IDO" - GOTO515
510 PRINT"[CLR][2CRSRD]##[SPC]ERROR-
HASESPCJPERDIDO"
515 NS=1:FORN=1TO1:NEXTN:IFX=4THENNM
=P
520 PRINT"(CR5RD1"NM-1"PASOS(SPC)FUE
RONESPC1CORRECTOS":PRINT"[CRSRD1[2SPC]
NO. ESPCIDEESPCIJUEGOS . "G
524 REM##QUINTA SINFONIA############
106.000
525 FORL=54272T054296 POKEL: 9 NEXT
538 POKEY, 15 POKER, 0 POKESR, 240 REST
ORE
535 READH IFH=-1THENFORL=54272T05429
6 POKEL , 0 NEXT - GOTO 600
540 REHDL - READD POKEHF , H . POKELF , L . PO
KEWA, 17 · IFNS=1THENPOKEWA, 129
545 FORT = 1TOD NEXT POKEWA, 16: IFNS = IT
HENROKEWA, 128
550 FORT+11050 NEXT-0010535
555 DATA25,177,100,25,177,100,25,177
 , 100, 20, 100, 800, 8, 0, 250
560 DATA22,227,108,22,227,108,22,227
 ,100,19,63,800,-1
565 REMARTERMINACION CORRECTA中非由中非中非
自由电
57@ PRINT"[CLR][2CRSRD]##[SPC]ENHORA
BUENA.[SPC]HAS[SPC]TERMINADO[9SPC][2CRSRD]
 [4SPC]CON"N1, "PUNTOS.
 575 PRINT"[CRSRD][25PC]NO.[SPC]DE[SPC]
 JUEGOS: "0:TT=TT+N1.PRINT"[CRSRD]##[SPC]
 PUNTUACION(SPC)TOTAL: "TT-NS=0
 580 GOTO525
 599 REM##VER LA ULTIMA SECUENCIA####
 the state of
 600 IFX=30RX=4THEN650
 605 IFHM=1AND(X=10RX=2)THEN650
 610 PRINT"[CRSRD]##[SPC]QUIERES[SPC]
 VERESPOJLACSPOJULTIMACI6SPOJCORSKDJC2SPOJ
 SECUENCIALSPOICS/N)?
 615 GETAS . IFAS= "THEN615
 620 IFHSO "S"ANDASO "N"THENG15
625 IFAS="S"THEN PRINT"[CLR]"BSCS
 630 IFH#="N"THEN650
 635 IFN=N1+1THENNM=NM+1
 640 FORN=1TONM-1-GUSUB400 NEXTN
 649 REM##QUIERES CONTINUMR?########
 直接施
 650 PRINT"[CLR][2CRSRD]#[SPC]QUELSPC]
 QUIERES(SPC)?(20SPC)[CRSRD]1.CONTINU
 ARTSPC116UHL"
 655 PRINTME3SPCJECRSRDJ2.CONTINUAR(SPC)
 CAMBIANDO(SPC)LAS(SPC)CONDICIONES(3SPC)
 [CRSRD13.TERMINAR.
 657 PRINT"(CRSRD]#[SPC]PULSA[SPC]EL[SPC]
 NUMERO"
 660 GETAS - IFAS=" "THEN660
 665 Y=VAL(A$): IFYO MANDYO 2HNDYO 3TH
 EN668
 670 DNYGOTO95,675,680
 675 GOTOZ®
 680 POKEY. 0 POKEWA. 0 END
 699 REM##FOR STACK CLEARING########
  療療施
  700 IFQ-1THENQ=0 GOT0500
  705 IFQ=2THENQ=0.00T0510
  718 IFQ=3THENG=0.GOT0570
```


EXCURSION POR LA MICROPROGRAMACION

Cursillo de Lenguaje Máquina —

Vamos al estudio del juego de instrucciones y veremos unas aplicaciones prácticas. Como diumos en el número anterior, se pueden según su tipo en instrucciones de carga y almacenamiento, transferencia, saltos, lôgicas y aritméticas.

Se denominan instrucciones de carga a las que tienen por misión "cargar" algo en un registro de la CPU (LDA, LDX y LDY en todos sus modos de direccionamiento). Las de almacenamiento son aquellas que "almacenan" o guardan el valor contenido en uno de los registros en una posición de memoria (STA, STX) STY). Las de transferencia "transfieren el contenido de un registro a otto CLAX, TAY, TSX, TXA, TNS v TY Vi Las de saltos son tres tipos JMP, JSR y todas las de "branch" BCC, BCS etc.) Lógicas son aquellas que realizan las "operaciones lógicas" (AND, EOR y ORA), de desplazamiento las que "desplazam" bits de algún dato (ASL, LSR, ROL y ROR) las de comparación "comparan" datos (CMP, CPX y CPY),) aritmèticas son las que efectuan este tipo de operaciones (ADC, DEC, DEX, DEY, INC, INX. INY y SBC).

Lambién podemos encontrai otros tipos de instrucciones que son las que afectan a las banderas del registro de estado (CLC, CLD, CL1 y CLV), y las que alectan al Stack (PHA, PHP, PLA y PI PI

Para saher que significan estas siglas sin tener que esperar a que aparezcan en la revista todas las tichas de los mnemónicos, aqui publicamos una tabla del juego de instrucciones en orden alla-

ADC Suma la memoria al acuhético: mulador con acarreo.

AND Hace "AND" con la memoria y el acumulador.

ASI Desplaza un bit a la



izquierda (memoria o acumulador)

Salta si carry=0 Salta si carry=1

BIQ Salta si el resultado es cero

Compara los bits de la memoria y el acumulador BILL

BMI Salia si el resultado es menus. BNI Salta si el resultado no es

Salta și el resultado es mas

BPL Salta si el resultado BRK Interrupción por software (break).

BVC Salia si no hay sobrecarga Salta si hay sobrecarga BVS

(1 (Hace C=0 (clear carry)

(11) Hace D=0 (clear decimal mode)

1=0 (clear interrupt disable bit)

Borra bit de sobrecarga CLV (V=0).

CMP Compara memoria y acumulador

Compara memoria e indice

CPY Compara memoria e indice

DEC Decrementa en uno la memoria.

DEX Decrementa en uno el indice X

DEY Decrementa en uno el

EOR "OR exclusiva" de la me-

moria con el acumulador Incrementa en uno la memoria.

Incrementa en uno el indice X.

INC

Incrementa en uno el in-INY dice Y.

Salta a una nueva direccion (como GOTO en Basic) MP

ISR Salta a una nueva direccion guardando la dirección de retorno (como GOSUB).

1 DA Carga el acumulador con la memoria.

1 DX Carga el indice X con la memoria

[1]) Y Carga el índice y con la memoria.

ISR Desplaza un bit a la derecha (memoria o acumula-

NOP No opera (no hace nada) ORA "OR" de la memoria con

el acumulador PHA Guarda el acumulador en el stack.

PHP Guarda el registro de estado en el stack

Saca el acumulador del stack.

PLP Saca el registro de estado del stack.

Gira un bit a la izquierda (memoria o acumulador) ROL

ROR Gira un bit a la derecha (memoria o acumulador).

Vuelve (return) de una interrupcion

Vuelve de subrutina (return)

SBC Resta la memoria del acumulador

Pone carry-1

SLD Pone modo decimal (D-1) St 1 Deshabilita las interrup-

ciones (l≈l) Almacena (guarda) el acu-

mulador en la memoria SIA Almacena el indice X en la

memoria.

Almacena el indice Y en la memoria.

IAX Transfiere el acumulador al indice X

سم

EXCURSION POR LA MICROPROGRAMACION

	INSTRUCTIONS	100	MED	LATE	1	AIISO	4011	1	£EH0	PAG	1	Apr	CUM		OLEY LE	EU	Ti-	IND :	1	cjd	101, 1	П	2.04	IGE D	Т	ABS.	1		MS T	Т	HELD.	TIVE	FI	ADINE	601	111	PAGE	1	PROC	E220H 1	OU STATUS				
e monic	OPERATION	ijΡ	7	-	+	_	_	-	1	$\overline{}$	\rightarrow	_	_	DP	7	_	OP		_			+	-	\neg	+	1	T	-	T .	-	$\overline{}$	$\overline{}$	+	T		_			CODE:	5 4	3 3	- 6	Louis		
		_	₩	1	1.	-	+	-	+	+	-	ar I		UP	-	4	₩	п	_	100	-	-	-+	1 1	+-	-	_	00	\vdash	-	1 90	n ir	90	n	,	DP.	n		N Y	B	Ďí	2 E	MMEA		
3.0	A+M+C-oA (4)(1)		2					- 1			2						6.	6		I_k		- 1		4 3		4		19		3									N Y			1 0	A		
N D	AMM +A	29	2	5	20			- 1		- 1	2						2.	6	2	3	5	2		4 3				39	4	3									N			1	RI		
5.	0				BE	E 6		٠ ١	26	۱ ۱	c 1	Dult 1	2 1				Н						6	ъ.	2 E		3												N			t c	A.		
4 6	BRANCH ON E×0 2)																1													5	10 2	2 4											a.		
[]	BRANCH ON LP. (2)	L	L	L	╄	\perp	\perp	4	_	4.				┸			1									L	ļ			Ŀ	30 4			L.									9		
Đ.	BRANCH ON 2+1 2										-								I			Ĭ			Т				П	- 1	0 .	1 2	Т			П							8		
	AAM				H	1		3	14	.	2											- [M M,			7	8		
MI	BRANCH ON N+1 27										-																			- 13	10 2	2 2											81		
N E	BRANCH ON Z=0 (2)						1				-																		1	1	00 2	7 2											В		
P	BRANCH ON NEU 21					1	1				-1																				10 2	7 .											ы		
RK	BREAK (See Fig. 1)	Г	Г	Т	Т	Т	Т	Т	П	Т	T	Т	T	00	7	1	П			_1	\Box	寸	\neg	7	\top		+	\vdash	П	\top			1	\vdash	\vdash	\vdash	-	-		1			В		
Y C	BRANCH ON Y=0 (2)										- 1						Ш														00 1	/ z											8		
V S	BRANCH ON V+1 (2)										- 1						Ш													- 1	D 2												В		
10	U -+C							-1			-			18	2		Ш																Н									D			
D	0 +0							-1		1				DB			Ш										,																		
4	0-40	Н	1		╁	+	+	+	+	+	+	+	+		2	_	Н	Н	\dashv	-	Н	+	+	+	+	+	-	┢	\vdash	+	+	+	╁			╁╌	Н	H	\vdash		D		1 €		
1 9	D +¥]											68		1	Ш																						Ď		U]		
M P	AM	LEA.	1,	١,	1.0	5 a		3 (3		,			100	1				,	Bi I	161	, ,	An	٠.	l lov	L		10	4	,			П										6.1		
ł 1	A M	į.e		1	E			- 1	4	- 1	- 1	ı	ı		ľ		"	ľ.	٠	D1		· [mo I	1	1	1	1	177		1			П						H			2 0			
	Y 64		1	1				- 1	74 I		- 1						П						- (H				()		
₽ =		r.o	7	1	+	\rightarrow	-	-	\rightarrow	-	\rightarrow	+	-	+-	-	-					-	+	-			1					-	-	-	1	_				H			<i>I</i> 0	-		
t L	M (+M		-		,F	1 1	' ·	1 1	.6	1	z			l	l.,							- 1	136	6	DH		3	-											N			ž.	ŢF.		
(X	ke K														2																								N			Į	į.		
6 4	71 97		١.	١.	١									80	1	h	Ш									1	1												N			J	D		
0.8	A+ MyA	49	1	1		0 4					3						8.	ь	٠.	51		- 1		4			1	59	4	1									N			7			
H	ld+ +ld		┡	\perp	E	i h		1 1	В	3 .	2	4		↓.	ļ.,					_	_	_ !	16	6 1	1		3	_		4		\perp	\perp		L.,				N			-1			
НΙ	Rel ex									1	1			6.0	4		Н																						ħ			t	,		
N F	tel-et			1				- 1						CI	2	1.1																							н			2	,		
HP.	JUMP TO NEW LOC				40		- :	3																			1						66	5	3										
. R	JUMP SUB (See Fig. 2)				70) h	١.	1				- }														,	4) ÷.		
D-4	M 4A	49	2	4	Į.AC	3 4	1	3 1	là		4						A.	h		8	5 .		85	4 2	B4.	4	. 3	E4	4	3									N			l			
Dκ	16 +1	42	2	Т	At	1		1 1	46		2			7									1	1		1		BL	1	3	7		\uparrow	\top		B6	4	7	ы			- 1			
D +	May 1	Ab	2	4	140	c 4	1 :	3 1	ы										-				84	ι,	8.	1													N			I	1.		
4	0+0 +0				140	6		3 4	là i		. 4	14 .	,										56	ь.	. Ы		3												g			1 .	1		
100	NO OPERATION													EA	4		'																										N		
184	AVM +A	09	1.	2	DE	0 0		1 0	25								١,	6	2	1	5	. 1	.5		10	. 4	,	.9	4					i					N			£	-01		
н1	Relit 2 1-5			+	+	+	+	\rightarrow	\rightarrow	\rightarrow	7	\rightarrow		48	,					-	-		-	+	+			+	\vdash	\rightarrow	+	+	+	\vdash		1-		-			_		P		
нР	PaMs 5 1+5				1									U8	1																												P		
F A	\$1, -05 Ms -44													68																									N			Į	P		
4	31 45 M5 44				П									28															H							Ī				duc	TORE D)		μ		
					1,				,					140	Ι,	1							34.	0	, 10	,	,												H	Fur 35	DATE OF	1 0			
10.		-	╀	╀	-	-	\rightarrow	\rightarrow	\rightarrow	-	\rightarrow	\rightarrow		+	+		Н	Н	\dashv	-	-	\rightarrow	\rightarrow	\rightarrow	-	\rightarrow	-	-	Н	+	+	+	+	⊢	-	\vdash	\vdash		-		_				
O R					58	6	1	5	363	7	4	ha .	4										r _b	D	14		- 3												N	474.5	OPCI	1 0			
1 1	RIRM INT (Sau Fig. 1)					1								40																			1							the 2	GUED)		R		
112	RINN SUB-See Fig. 2)													60	6																												R		
S & C	A-M-C-sià (I)	19	1	2	140	1	1	1	15	1	1						E.	Б	4	F1	5		15	٠.	' '	4		19	ě.	3									el A			2 3	3		
Ev	∌ €													36		1																											\$		
0	, eD													FB	+	٠								4	1					_			1								4		,		
.1	-91				1									718	2	1																				1							5		
à	A-mid				ļ.	0 4		3 1	15		2						5.	Ę.	2	91	6	2	95	4 3	90	5	. 3	99	5	3													5		
13	K-eld				85	1	1	3 (16		2																									96	4	2					S		
1 4	Table				di)	i 4		3 1	н :	1	4												94	4 3																			S		
4.3	Re-A													Alt	2																					1.			N _			Ł	17		
17	A-a-Y									1				AB	2	Γ.																	T	1	13				N			l.	Ti		
	A-pY													BA																									N			ı	1.5		
K Z	5-eX													BA		1					-																		lr			1	1		
# A	K-eA															١,																											10		
3 S	1-65													9.1	2	١,																							N			ı	1 11		
Y Ř	Yesh													98	2	,																													
	(I) ADD L to "N" IF P	LGE	BOUI	HOAR	Y IS	CRO	SSEC	0													1	13	1364	JI									400					М	M	ILWORY	BH3	7			
	(2) ADO L TO "NO F 6	RAN	ен е	COUR	RS TI	O SAI	ME P	PAGE													¥		NDER										SUB	TRAC	T			M	u M	IEMORY	803	6			
	A00-2 TO "N" IF 6			NUUI	K) FI	o (III)	PEN	1,161	- MGE												A	A	IÇEUN	lubit	DIF							J.	AHC)				16		8 00					
	(3) CARRY NOT - BOR			e ie i	Inva.	i in															М	М	AE MOI	RY PE	EFTE	CI WE	ADDR	SE3S				F	90					- 0	20"	O BY!	£S				
	14) IF IN DECIMAL MO																																	, SIY											



EXCURSION POR LA MICROPROGRAMACION

puntero del stack. Transfiere el indice Y al

Puede parecer rara e inútil la instrucción NOP, ya que no hace absolutamente nada, pero esta instrucción es muy útil en contra de lo que se puede suponer. Sus aplicaciones principales son las de introducir un pequeño retardo en algún lugar de un programa, y su función más importante es en el desarrollo de programas, cuando todavia no hemos conseguido una versión definitiva y seguramente se nos olvida alguna instrucción importante, si hemos colocado unas cuantas instrucciones NOP por aqui y por allá, bastará con cambiarlas por las instrucciones que olvidamos, con esto nos ahorraremos tener que reescribir todo el programa.

TAY Transfiere el acumulador al indice Y.
TSX Transfiere el puntero del stack al índice X.
TXA Transfiere el indice X acumulador.
TXS Transfiere el indice X puntero del stack.
TYA Transfiere el indice Y acumulador.
Puede parecer rara e indica instrucción NOP, ya que no absolutamente nada, perc instrucción es muy útil en de lo que se puede supon aplicaciones principales de introducir un pequeño en algún lugar de un pro su función más importa el desarrollo de prograr do todavía no hemos a una versión definitivamente se nos olvida trucción importanta colocado unas cua ciones NOP por actiones NOP por actiones NOP por actiones NOP por actiones nos ahorrar reescribir todo el Quizás algunc gados se pregur puedes interca nes que olvide otra línea intiquina no se acyplicación los saltos se de línea, y disminum sigue sier alteremos que ocul los salty memori del prelas ine lugar adeco regla (retique ocu) di fi Quizas alguno de los recién llegados se pregunte cómo en basic puedes intercalar las instrucciones que olvides con sólo escribir otra linea intermedia, y en máquina no se puede hacer esto, la explicación es sencilla: en basic los saltos se hacen a un número de línea, y cuando añadimos o disminuimos algo del número sigue siendo el mismo aunque alteremos la posición de memoria que ocupa. En código máquina los saltos son a direcciones de memoria, y si desplazamos parte del programa al intercalar algo, las instrucciones se moverán de lugar y los saltos no serán los adecuados. La excepción de esta regla son las rutinas reubicables (relocatable), que son aquellas que no contiene este tipo de saltos y que pueden cambiar de dirección sin afectar su correcto

Vamos a ver ejemplos sencillos de programas en código máquina, el primero de ellos reliena una zona de memoria con un dato.

START LDA DATO

STA DESTINO,X

BNE LAZO Esto en basic seria:

10 A=DATO

30 POKE (DESTINO+X), DATO

40 X=X-1

50 IF X <> 0 THEN 30 60 END

Este programa podria hacerse más sencillo con un lazo FOR/-NEXT, pero lo he escrito de este modo para que sea idéntico al de

código máquina. Una aclaración, en realidad el programa que aparece aqui no está en código máquina, está en lenguaje ensamblador, y necesitaremos realizar el proceso de ensamblado (pasarlo a c.m.) por alguno de los procedimientos posibles:

El más sencillo pero más caro es utilizar un macro-ensamblador (el C64 dispone de uno muy bueno que permite la utilización de etiquetas, detecta errores, etc.). Para el VIC-20 existen cartuchos como el monitor de lenguaje maquina, que facilita el ensamblado pero no admite la utilización de etiquetas.

Por último se puede realizar el proceso a mano, este es el método más didáctico pero al mismo tiempo es el más pesado e inhumano, consiste en ir buscando las instrucciones una por una en las tablas de mnemónicos, mirar en la columna del modo de direccionamiento que vamos a utilizar y colocar el dato en la memoria utilizando instrucciones POKE después de haber convertido el valor hexadecimal en

Respecto al modo de utilizar decimal. los ensambladores, no lo voy a explicar ya que los manuales respectivos dan todas las instrucciones para su funcionamiento y varian bastante de uno a otro.

Vamos por el camino dificil, lo haremos a mano:

Primero escogemos una posición de memoria RAM que esté libre. Como la rutina es corta podemos utilizar el buffer de cassete que se encuentra en \$033C=828.

Decidimos que el dato sea \$20 y que el destino será \$0400 (pantalla en el C64).

-Buscamos LDA en las tablas de mnemónicos y vemos que en modo inmediato es A9, por tanto LDA #\$20 será A9 20.

-Seguimos buscando el resto de las instrucciones de igual modo, pero al llegar a BNE LAZO tenemos un problema al hacerlo a mano, debemos calcular la "distancia del salto", para ello restamos la dirección de la instrucción siguiente a la dirección de destino y ese es el salto.

-Cuando terminemos con todo

vamos colocando los datos en las posiciones 828 y siguientes, y luego podemos probar la rutina con SYS 828.

Si el ensamblado lo hemos realizado correctamente, debemos borrar la parte alta de la pantalla en un instante. Si queremos borrar menos caracteres podemos cambiar el número \$FF en LDX #\$FF para que borre lo que queramos.

En código máquina la rutina

debe queramos. LDA#\$20 033C A9 20 033E A2 FF LDX#\$FF STA\$0400,X 0340 9D 00 04 DEX 0343 CA BNES0340 0344 D0 FA RTS

Para el VIC-20 debemos cam-0346 60 biar el STA\$0400, X, por la dirección donde se encuentre la pantalia, \$1E00 si no tiene ampliación RAM, que seria 9D 00.

La columna de la izquierda del listado es la dirección donde lo colocamos (en hexadecimal). Las del centro son el código de operación y el operando, que es lo que debemos colocar en la memoria para decirle al ordenador que haga "eso", y lo de la derecha son los mnemónicos. Si disponemos de un monitor de lenguaje máquina bastará teclear la columna de la izquierda y las del centro, con un ensamblador entrariamos la de la izquierda sólo la primera vez, y luego la de la derecha, ya que el mismo calcularía la siguiente dirección (columna de la izquierda).

Si tenemos que entrar los datos sin ayudas desde el basic, convertiremos los datos en decimal (169, 32, 162, 255, 157, 0, 4, 202, 208, 250, 96), y los introducimos mediante POKE en modo directo o colocándolos en sentencias DATA y con un lazo FOR, READ, POKE, NEXT, o similar. Como ejemplo, aqui tenéis un cargador para este programa:

10 READ A:1F A=-1 THEN

20 POKE 828 + Z, A:Z = Z + 1:GOTO 10

100 DATA 169, 32, 162, 255, 157, 0, 4, 202, 208, 250, 96,

Para probarlo ya sabeis, RUN para colocarlo en sus posiciones y luego SYS828 y RETURN para ejecutarlo.

Si queréis podéis probar su equivalente en basic y comparar los tiempos.

M/A/R/K/E/T/C/L/U/B

Servicio gratuito para nuestros lectores particulares. Empresas 300 ptas, por línea, De no especificar duración, los anuncios serán publicados durante 1 mes. Debido a ciertos abusos que nos han sido comunicados, los anuncios gratuitos de Market club solamente serán publicados si vienen con nombre, apellidos y dirección completa. No se admitirán apartados postales. Lamentamos que todos tengan que sufrir esta medida por culpa de algunos "listos Adicionalmente es recomendable que enviéis el número de teléfono por si hubiese alguna pega, si no deseáis que lo publiquemos podéis decirlo en la carta.

MERCADILLO

 Vendo un ordenador Commodore 4000 y otro 3,000. Buen estado. Con programas. Teléfono: (91) 402 38 89 / 401 21 50 (Ref. M-70)
 Vendo VIC-20, Cassete C2N, Placa expansora 4 cartuchos, Curso completo de Basic (2 libros y 4 cintas), Guia de Referencia al Programador VIC-20 y 15 tas), Guia de Referencia al Programador VIC-20 y 15 juegos comerciales por 35.000 ptas. (José Lainez, Telefono: 610 70 80 de 15,30 en adelante) Todo ello comprado en julio de 1982. (Ref. M-86) el Intercambio programas para C-64 en cassette, Julio Rodriguez, C/Zurbano, 93, 28003 Madrid. Telef.; (91) 234 98 98. (Ref. M-87) el Se vende late de más de 150 geograpas por 5.000.

(91) 234 98 98. (Ref. M-87)

• Se vende lote de más de 150 programas por 5.000

• se vende lote de más de 150 programas por 5.000

• secribir a: C/Mallorca, 420, 6° 1°. 08013 Barcelona, o

a Xavier Querol Carrillo, C/Córcega, 613, 5° 1°.

08025 Barcelona. Mandamos lista. (Ref. M-88)

• Vendo VIC-20 con 32Kb de RAM, código máquina, 2 cartuchos de juegoa, 1 programa de

fichero y otro de agenda, curso Basic primera parte, guia de referencia del programador, y mucha infor-mación y revistas. Urge vender por cambio de equipo, Juan Allimira Vilar, C/Pintor Togores, 12, 3º 1, Cer-danyola (Barcelona). Teléf., 93-691 07 28. (Ref. M-89), Vendo programa en cinta VIC-20, Ajedrez (nueve niveles); Vic print (procesador texto); Vic calc (hoja cálculo); Graphics (alta resolución); odisea espacial cliculo); Graphics (attal resolution); disca espacial (juegos); Tienes retentiva (educativo); I* parte programación + libro todas por 7.000 ptas. Carlos Infante. Tel. (93) 311 27 06 (Barcelona); (Ref. M-90) • Si alguien ha copiado y vende los programas listados en los dos libros de "curso de introducción al Basic" de Andrew Colin editado por Micro Electró-nica, y desca ganarse unas perras dando servicio a los demás, espero que se ponga en contacto conmigo bara venderme una copia, ya que quiero introducir a mis hijos con ayuda de estos libros, pero no tengo tiempo. Alfredo Ruiz Martínez, C/Juan Gálvez, 18.

41007 Sevilla (Ref. M-91)

• Vendo un VIC-20, 24,000 ptas. Memoria de 16K 9900 pias., 5 juegos Indescomp a 800 cada uno, 2 juegos ingleses a 900 cada uno, 1 juego en cartucho 1,300 ptas; 1 libro con 30 juegos, 1,200 ptas., 1 libro en inglés, 1,200; 50 programas escritos a máquina 1,000, o todo por 36,900. Llamar al 446 65 56 de Madrid. Preguntar por Ismael (a partir de las 16:30)

(Ref. M-92).

(Ref. M-92).

Me gustaria cambiar programas y juegos para el VIC-20. Tengo super-expander y cartucho de 16K. Llumad a Jose al 93-422 66 41 (casa) o al 93-332 56 90 (trabajo) Barcelona. (Ref. M-93)

 Cambio controlador doméstico de 8 reles y cartucho de juego ROADRACE por impresora para el VIC y si parece poco también regalo ZX-81 con 16K + juegos con manuales, etc. También cambiaria el ZX-81 con todos sus complementos por superexpander o cartucho de 16K. José Tomás Gómez. C/Maria Díaz de Haro, 16, 3-F. Bilbao (Vizcaya).

6 Descaria intercambiar juegos y todo tipo de programas para el C-64. Escribir a: Eduardo Martin Ruiz, C'Almagro Segura, 4. 18012 Granada, mandando lista de programas, o tlamadme al Teléf.; 282 28 22 60. Interesan sobre todo juegos. (Ref. M-95)

 Estoy interesado en el intercambio de programas de juegos para el C-64. Escribir a: Alejandro Canto Navarro, C/Doctor Gadea, 1, 2º E+F, 03003 Alicante o llamar al teléfono 965/22 33 30. (Ref. M-96).
 Estoy jinteresadisimo!! en el cambio y compra de juegos y programas de todo tipo del C-64 en cinta, luan Martínez López C/Honduras G-59 G-09 19. Juan Martinez López, C/Honduras, G-5°, G-9°, 1°, 1°, A. Barcelons (27). Tel. (93) 340 94 65 (Ref. M-97). Desearia intercambiar programas para Commodore 64 con usuarios del mismo. Soy Juan Carlos Stubbs Cruz, C/Mirador, 8 "ALBION" Urbanización "Cerrado de Calderón" 29018 Málaga, Telefono: 952/29 15 74. (Ref. M-98)

 CBM-64 Intercambio o vendo programas comerciales. Gonzalo Machado Gallas, c/Marin Ocete, 8, 69 18014 Granada. (Ref. M-99)

Compro programas para C-64 juegos y aplicaciones. En especial ajedrez, trazado de funciones, esta-distica, etc. Programas de gestión. Todos en cinta o cartucho, Jordi Sierras. Lista de correos. Alella (Bar-celona). Tel. 93-555 22 64 (Ref. M-199)

• Cambio cualquier tipo de programas y experiencias para el C-64. Sergi Arbos Mauri, C/Forn del Vidre, 16, 3", 4", Vitanova i la Gettrú (Barcelona). Tel.: 93-893 18 31 (Ref. M-101)

· Vendo microordenador Dragon 32 por estrenar en perfecto estado, con cables para conectarlo a cual-quier cassette y a TV. Regalo 2 juegos en cinta. Manual en castellano, Precio a convenir. Interesados

dirigirse a: Eusebio Zuloaga Arisa, C/Balmes 444, 2º 1º 08022 Barcelona. Tel.: 93-247 22 76 (Ref. M-182)
• Tengo 15 años y el VIC-20, mis posibilidades económicas no me llegan para adquirir un periférico, por esto agradeceria si alguien me pudiera vender el Datassette por unas 5.000 ptas. Mis señas son: David Puig Rojals, c/Lorenzana, 36, 40 B. Gerona. Tel. 972-

20 46 3 L (Ref. M-103)

 Vendo o cambio cartucho de juego ref. VIC-1097 "Jupiter Lander" por cartucho de ampliación Ram de 8K o 3K También Introducción al lenguaje de programación Basic por 3000 ptas. Desearia contactar para intercambio de programas del VIC-20. Francisco Fernández Iglesias, C/Ramiranes, 4, Porrino (Ponte-vedra) (Ref. M-104)

 Vendo expansor de 4 cartuchos para el VIC-20 marca indescomp con interruptores individuales por 6.000 ptas y cartucho lenguaje máquina VIC-20 por 3,000. Intercambio programas para C-64 cints y disco. Miguel Angel Martinez Martinez, C/Madre de Dios, 16, 4º-D, Izda. 26004 Logroño. La Rtoja.

● Impresora MPS-801 Commodore vendo por 45,000 ptas, (comprada en mayo 1984) por cambio a modelo mayor. Interesados dirigirse a: Manuel Vijande, Tel.: (985) 25 21 56 de 8 a 10, mañanas y tardes. (Ref. M-106) Compraria impresora para CBM64 precio a conve nir. ¿Acabas de comprar un C-64 y no sabes cómo usarlo? Vendo la mejor introducción al Basic, sencilla, práctica y fácil de entender: EL CURSO DE INTRODUCCION AL BASIC, PARTE I, de Andrew Colin, Este libro tiene cantidad de ejemplos y programas a probar, muy fáciles de entender. Lo vendo por 1,850 ptas. Mi teléfono es el 229 07 02 de Barcelona, Preguntar por Francesc, Francesc Guasch Ortiz, C/Jerez nº 40, 08032 Barcelona, (Ref. M-107)

Por cambio de ordenador vendo ampliación de 16K Por cambio de ordenador vendo ampliación de 16K (13.000 ptas.); video-juego en cartucho ALIEN (3.600 ptas.); cartucho de ajedrez SARGON II CHESS (3.600 ptas.); cartucho SUPER EXPANDER (5.800 ptas.) Todo junto por 20.000 ptas. Manuel Fuentes Sorrivas. Apartado de Correos, 1.473. Vigo (Pontevedra). Teléfi: (986) 47 17 68. (Ref. M-108)
 Vendo 3 cassettes de juegos para el VIC-20 por 1.000 ptas. cada uno. Los juegos son: Tron, Skramble y Odisea Espacial (16K). Vendo también la Guia de referencia del programador del VIC-20 y el curso de referencia del programador del VIC-20 y el curso de

referencia del programador del VIC-20 y el curso de introducción al basic 1º parte, cada una por 1.300 ptas. Llamar en horas de comida o cena al telét.: (985) 56 13 90. O escribir a Benito Martinez Fernández. Colonia del Quirinal, nº 3, 1º izda. Aviles (Asturias) (Ref. M-109)

Vendo dos programas para el C-64: "Shadowfax" "Compiler", 3.000 ptas los dos. Agustin Quevedo Velasco, C/Oña, 55, 4º B. Madrid Tel.: 202 94 28

(noches) (Ref. M-110)

Cambio 2 programas de VIC-20 por 2 de C-64 hechos por equivocación. Son Topo loco y Tron. Más información sobre los programas en Avenida del Norte nº 8 atc. 3º Manuel González Martinez. Hospi-talet (Barcelona). Teléf.: 240 23 50. Llamar sábados por la tarde, Domingos todo el día. Entre semana de 2 a 3 del mediodia (Ref. M-111) Deseo ponerme en contacto con socios o colegas radioaficionados para intercambiar programas y experiencias. Tengo un C-64. José Luis Espinar, Indicativo: EA7EBT. Apartado 222. Fuengirola, Málaga. Teléf.: (952) 47 10 76. de 15 a 22 horas (Ref. M-112).

• Vendo VIC-20 con: Superexpander +3Kb, Vicmon (monitor leng, máquina), vic spedd (graba 6 veces (monitor ieng, maquina), vic spedd (grana 6 veces más rápido), y más de 150 programas (comecocos, Defensa, Traxx, Vic-calc, Vic-Base, etc.). Francisco Plaza Pérez C/Jesus Rivero Meneses, 2, 47014 Valladolid. Teléf.: 33 23 25. (Ref. M-113)

· Vendo C-64, nuevo, sin uso, comprado en noviem-• Vendo C-64, nuevo, sin uso, comprado en noviembre del 83, con cassette Commodore, comprado en febrero del 84, con libro "guia de referencia" Precio real 94,800, precio de venta: 75.000 ptas. Luis Javier Redondo Fernández, C/Nicasio Pérez mim. 9-3º bis 47005 Valladolid. Teléf.: (983) 29 93 84. (de 11 a 11,30 noches). (Ref. M-114).

• Vendo VIC-20 comprado en enero 1984 por 25.000

ptas, cassette Commodore por 9,000 ptas; 5 juegos de cinta Indescomp y 1 de cartucho Commodore por 14.000 ptas; Gula referencia programador y Busic I con cinta por 1.000 ptas. Comprando todo el lote haria más rebaja y regalaria dos colecciones de revistas, joystick y cassette grabado con [1] juegos. Llamen o escriban al teléf.: (91) 613 49 44, Antonio Aguilar Melian, Parque Zarauz Bloque 4º Mostoles (Madrid). (Ref. M-115)

Vendo y cambio programas para C-64. Base de datos, Ajedrez, Música, juegos, etc. Jordi Bachs, Rambla San Sebastián, 30 4º 1º, Sta. Coloma de

Gramanet, (Barcelona) (Ref. M-116),

Vendo VIC-20 con expansión de memoria 16K más de 100 programas diferentes y de calidad: Juegos, utilidades, base de datos, etc. cartuchos y específicos lidades, base de datos, etc. cartuchos y específicos para radioaficionados, junto a una extensa documentación y libros. Todo 40.000 ptas. Alfonso Moraleda Pérez. Clara del Rey, 81. 28002 Madrid. Tel.; (91) 415 02 24 - Noches. (Ref. M-117).

• Vendo C-64, unidad disco 154, juego joystic. Apto gestión. Nuevo. Buen precio. Escribir a Ricardo Soteras Sogas, C/Tres Creus, 79. Sabadell (Barcelona) Dar nº Telefono. (Ref. M-118).

• Desertia contratar con variatios de C-64, más o

• Desearia contactar con usuarios de C-64, más o menos de mi edad (14 años), para intercambio de programas tanto de juegos como de utilidades (sobre todo de matemáticas). Francisco Javier Almellones, C/Bahia de Algeciras, 8-3º C. Algeciras (Cádiz). C/Bahia de Algeciras, 8-3º Teléf.: 66 49 34 (Ref. M-119).

 Quisiera intercambiar juegos con otros lectores de esta magnifica revista. Mi dirección es: Luis Alfonso Sanz, Pº San Isidro, 5, 2º A. Valladolid-12. (Ref. M-120) Vendo Ordenador Personal Commodore 8032.
 32K, con Floppy doble 4040, nuevo. Precio total 300.000 ptas. Vendo también PET 2001, 7K, con cas-

sette y pantalla incorporados. Precio 40,000 ptas. Teléf.: 93-674 62 33 (horas de cenar) 6 93/661 55 16 (mañanas). Barcelona (Ref. M-121)

 Por cambio de equipo vendo VIC-20 comprado en noviembre del 83, totalmente nuevo. Regalo al comprador más de 100 programas para VIC básico o con ampliación de memoria (muchos de ellos comerciales y de utilidades, como VIC-PRINT, VIC-CALC, Bases de datos, ajedrez y juegos diversos). Y sólo por

CENTRO DE INFORMATICA

Las Rozas - Majadahonda **EMPEZAMOS** Cursillos en BASIC cada 15 días

Directamente en ordenadores VIC-20, COMMODORE 64 **SPECTRAVIDEO**

Teléfono: 637 31 51

25.000 ptas. Acepto ofertas. Dirigirse a Victor J. Calvo Medina. Calle Beatas 20, 3º D. 29008 Málaga. Teléfi. 21 44 09. (Ref. M-122).

Vendo VIC-20, casi por estrenar por 39.000 ptas.

con cassette manual del usuario, curso Basic I (libro y 2 cintas), cartucho de ajedrez, 2 cintas juegos de Indescomp y varias cintas más con 50 programas y juegos. Preguntar por Juan (93) 371 90 03 o escribir al apartado 37 de Esplugues (Barcelona). (Ref. M-123).

apariado 37 de Espugues (Barcelona), (182), in-180).

Cambiaria toda clase de programas para VIC-20 con o sin superexpander. Tengo algunos muy buenos. También los venderla, Alvaro Ibáñez Blas. C/Cardenal Silíceo, 20, 2º E. Madrid-2. Tíno.: (91) 416 09 13

 Compraría impresora VC-1525 o SEYKOSA, precio a convenir, si es posible que el vendedor sea de la provincia de La Coruña o cercanias también me interesa el monitor de lenguaje máquina. Para ofertas escribir a José Luis Ferreiro Meijide, Apartado 378;

escribir à Jose Lais retretto vivillate, Apartado de La Coruña, (Ref. M-125)

• Vendo C-64 por 50,000 ptas. Lote de programas Grandmaster, Base de datos, Gráficos 64, Monopoli, Rox 64, Tank Atack, Hustler, Frogger, Motormania, Control and Control Flight) todos en 5.000 ptas, Sueltos a 1.000 ptas. Controlador para controlar aparatos externos con 8 relés

por 6.000 ptas. Antonio Toral Peinado. C/Puente nº 3 Illora (Granada) Telf.: 958/46 30 09. (Ref. M-126). • Vendo ampliación de 16 RAM para VIC-20 por 12.000 ptas. Razón: Teléf.: 253 09 87. Barcelona. (Ref.

M-1271

· Vendo programa para C-64, 16 juegos y/o ampliaciones por cinta. 800 ptas, por cinta más gastos de envio. Razón: Maite, C/Moncada nº 5, 2º izda, dcha. Tíno.: (94) 443 67 79. Escribir o llamar. (Ref. M-128).

· Compraria programa Assembler (Ensamblador y desensamblador) en cassette para CBM 3000 con instrucciones. Ariemio González Pérez-Fernando Aro-cena Quintero, I. 38009 Santa Cruz de Tenerife. (Teléf.: 22 42 44). (Ref. M-129)

· Vendo NEW BRAIN 32K con display (cables, alimentación, cassette sanyo, manual, quia del principiante, technical software manual, programas) en perfecto estado por 65.000 ptas. Angel Osinaga Rubio. C/José Jimeno 2-2º Drcha, Pamplona 31015 Navarra, Teléf.: (948) 11 35 07. (Ref. M-130).

· Busco gente en Sevilla para poder comprar entre bastante gente la guia de programación (que son 2.000 ptas, del C-64) y usar luego el arte de la fotoco-2.090 ptas, del C-64) y usar luego el arte de la fotocopia. También busco gente con un nivel medio para compartir programas. Luis Luna Barrios. P/Gomila, 16-10. Sevilla, 7. Tfno.: 954, 51 33 45. (Ref. M-131).
Vendo unidad de disco mod. 1541 con sólo seis meses y muy poco uso por Ptas. 75.000. Interesados escribir a: Alfredo Ruiz Martinez, C/Juan Gálvez, 18, 41007. Sevilla. (Ref. M-132).
Vendo acon remendo de sudemodos. 15 febulosco acon.

 Vendo por cambio de ordenador, 15 fabulosos programas para cbm 64: Simons, Basic, Vic Tree, Fort Apocalypse, Choplifter, Solo Flight, Skramble New, Falcon Patrol, Moon Buggy, Pitstop, Soccer 64, Burgertime, Superpipeline, Radar Rat Race, Matria y Laser Zone por 8.000 ptas. Su precio de mercado es 10 veces. Urgente. Fernando. Horas comida. Tíno.: (91) 415 37 32. (Ref. M-1.33).
 Vendo VIC-20 comprado en dic-83, con el datasette, ampliación 16K, joystick, 5 programas y fibros. Todo 45.000 (De no estar de acuerdo con el precio hacer ofertas). Telét.: (976) 31 01 17 de Zaragoza · Vendo por cambio de ordenador, 15 fabulosos pro-

hacer ofertas). Teléf.: (976) 31 01 17 de Zaragoza Javier Boira. Via Ibérica-38, 8º Zaragoza. (Ref. M-134)

 Me interesaria conocer conocer si hay algún club de amigos del VIC-20 cerca de mi ciudad. Agustin Ruiz Castrillo. C/Gregorio Solabarrieta, 49 Miranda de Ebro (Burgos). (Ref. C-21).

Estamos formando un Club de Usarios en el centro

de Madrid. Cualquier interesado que se ponga en contacto con nosotros en el apartado 427 de Madrid, diri-giéndose a Rafael (EB4XD). Teléf. 276 66 94 (Dejar teléfono. Contestador). (Ref. C-22).

 Me gustaría entrar en contacto con usuarios del CBM-64, para intercambio de programas e ideas, asi como para trabajar en grupo". Antonio Iglesias Arias, C/San Isidro Labrador, 16, 1º B. Palma de Mallorca, 5. Baleares, Teléf.: 412 728. (Ref. C-23).

 Interesados en la formación de un club de usuarios de los ordenadores "VIC-20 y Commodore 64". Escribir o llamar a Félix Portabella Padro. C/Forn de Santa Llucía nº 1, 2º Manresa. Teléf.: 872 22 97. [Ref.

 Agradecería la dirección o teléfono de algún socio de mi ciudad o alrededores, a fin de ponerme en contacto con él para ver la posibilidad de formar un club o similar. José Esteban Llusar Molés, Daoiz, Daoiz,

2-1º 2º. Pto de Sagunto (Valencia). (Ref. C-34)

• Grupo de usuarios de Commodore en Tarragona, estamos intentando formar un club lo más sólido posible, no nos mueve ningún interés lucrativo, y estamos desarrollando un sinfin de actividades, encaminadas a tener un más amplio conocimiento de nuestro ordenador, Alfred Blasi. Tel.: (977) 547 307. Tarragona. (Ref. C-35).

 Deseo contactar con niños-as que deseen formar un club de ordenadores Commodore 64 y VIC-20 (puede ser otra marca), para intercambiar experiencias. C/Avda, del Norte nº 8 atc. 3º Hospitalet (Barcelona), Manuel González Martinez, Tel.: 93-240 23 50.

 Interesados en la formación de un club de informática en Granada capital dirijanse por correo a C/ San Jerónimo, 23, 2º. 18001 Granada, Miguel A, Puerta Carrasco, (Ref. C-28).

 Si queréis que hagamos un club "Vic-20" poneros en contacto con Antonio Recober. Teléf.: 32 20 46. Málaga. Se trata de intercambiar programas. Conocimientos y experiencia sin que nos cueste dinero. Si algún 64 quiere hacer el camino a nuestro lado. Podemos intentarlo. (Ref. C-29).

 Desearía contactar con interesados en formar un club de usuarios del C-64 en Granada e intercambiar programas y experiencias. Antonio Toral Peinado. C/Puente nº 3. Illora (Granada), Tel.: 958-46 30 09 (Ref. C-30).

Para formación club nacional de CBM Series 2000 y 3000 exclusivamente, ponerse en contacto por carta con Artemio González Pérez, Fernando Arocena Quintero, 1. Teléf.: 22 42 44, 38009 Santa Cruz de Tenerife. (Ref. C-31).

Me gustaria contactar con alguien de Motril para compartir experiencias y/o formar un club de C-64 (y VIC-20 si hace falta). Ya tengo local. Mi teléfono:

958-600 314 (Ref. C-32).

• Ha nacido el club "Brian & Chip" para usuarios del C-64 y Spectrum 16K. Crearemos una revista propia, pero necesitamos colaboraciones, programas e ideas. intercambiamos programas listados o en cinta, escribir desde cualquier rincon de España, prometo con-testar. Francisco Reig López, C/Marina, 261, 3° 5°, 08025 Barcelona. (Ref. C-33).

(Viene de pág. 4)

que podéis imaginar lo que supone una particula de polvo en el recinto donde lo fabrican.

La causa de que los chips pongan "made in Japan" es que allí fabrican las cajas al ser más barata la mano de obra, pero lo de dentro es 'made in USA". En fin que aprendimos bastante este día. Después la limusine nos llevó al aeropuerto de Filadelfia para tomar el avión de

vuelta a Nueva York.

El miércoles por la mañana tomamos otro avión desde el aeropuerto de La Guardia a Boston, donde nos recogió Scott Baum, que es el analista informático de los PCs en la sede de nuestra compañía en Framingham (MA). Nos llevó a comer y fuimos a visitar el edificio de la compañía, conocimos a Marion Kibbee, Debra Wetherbee y todos los jeses de los distintos departamentos, que nos enseñaron cómo atienden a la correspondencia, todo lo hacen con procesadores de texto igual que nosotros utilizamos el easy script, la diferencia estriba en la cantidad de personas que se encargan de la correspondencia y el volumen de esta.

Stott nos llevó de vuelta a Boston, por donde paseamos para ver los puntos interesantes y luego fuimos a cenar en un relajador restaurante situado en uno de los edificios más altos

de la ciudad con vistas al puerto. La cena, la compañía y las panorámicas fueron encan-

Al día siguiente llegó Olga Greenwood con el conductor de la limusine, que era similar a la de Commodore pero mejor (además de lo de la otra tenía video para ver películas en el camino..., nos llevó a la sede de la compañía en Peterborough, donde hacen la hermana gemela de la Commodore World en América, que se llama RUN. Alli fuimos a comer con Guy Wright (el editor técnico de RUN), Dennis Brisson (el editor jefe), Matthew Smith y Michele Christian. Olga nos prestó una gran ayuda, ya que hablaba perfectamente español, y puedo asegurar que pasar varios dias seguidos traduciendo todas las explicaciones sobre las fábricas de Commodore y circuitos integrados, en las compras y en todo, es una tarea que da bastantes dolores de cabeza, ya que la única persona con la que hablaba mi idioma era con Joan, y para él también fue un alivio poder hablar con otra persona. Luego dimos un paseo en un coche de caballos viendo los frondosos bosques, el lago, y todos los rincones pintorescos incluyendo el rancho donde Michele guarda un potrillo joven. La limusine nos recogió en el rancho para llevarnos a visitar las instalaciones de RUN, 73,s, Hot Co-co, 80 Micro, Incider, etc. Le enseñaron a Joan el proceso de fabricación de la revista desde el principio al fin. Parecian surgir los computadores en todas las fases de la revista, era fantástico. Más tarde volvimos al hotel en Boston y dimos una vuelta por las instalaciones deportivas antes de cenar, Joan prefirió verlo todo, pero a mi me gusta más la natación y aproveché la oportunidad en el hotel, luego cenamos y dimos una vuelta por el Quincy Market que es uno de los lugares típicos de esta ciudad.

El viernes teniamos planeada una visita al acuarium y museos de la ciudad, pero el cansancio hizo que la anulásemos, nos fuimos a dar una pequeña vuelta por la ciudad para hacer las compras de rigor (postales y cosas de esas) y tomamos un avión hacia Nueva York.

Tras todas estas peripecias volvimos al hotel, y el sábado dia libre para compras..., también nos recorrimos la parte sur de Manhatan, teniamos que aprovechar el día ya que se terminaba el tiempo y queriamos verlo todo.

El domingo fuimos apenados a tomar el vuelo de vuelta, y unas horas después en casa otra vez, el sueño dorado había terminado y la vida normal se imponía, yo de nuevo a Madrid y Joan a sus montañas pirenaicas.

COMENTARIOS COMMODORE

(Viene de pág. 25)

unidades de discos, para recuperar estos datos han añadido un utilitario llamado "regeneración/verificación disco", que se encarga de recuperar los errores producidos si por accidente se destruye parte del disco, También para evitar problemas se han utilizado ficheros de acceso directo del tipo USR, en lugar de los relativos

De todos estos detalles que he mencionado. no tenemos que preocuparnos, ya que a nosotros nos pregunta si queremos copiar el disco, entrar asientos, etc. Todos los detalles técnicos los maneja el programa por si mismo.

Los Ficheros URS son complicados para el programador, pero evitan errores en los discos y como el programa ya está hecho no nos importa su complicación. El dividir el programa en varios programas, permite no saturar la memoria, con lo que la rutina de "limpieza" se necesita menos veces ganando velocidad y permitiendo que todo el programa sea enorme aunque esté dividido en pequeños bloques. El único aspecto negativo es el pequeño tiempo que se pierde cuando el propio programa carga otra parte, pero creo que merece la pena,

Por último podemos decir que se adapta bien al plan contable español, y yo creo que es un buen paquete de software.

CLUBS

 Club de amigos. Foto Estudio 2. Plaza de Sombre reros, 2, Palma de Mallorca, Tel.: 21 31 62. (Ref. C-1)

Club de programación Alaiz. Tels.: 254 480-257 704.

Pamplona. (Ref. C-4) Clitb de Commodore de Albacete. Fernando Martinez Guerrero. La Roda, 39. 5st D. Albacete. (Ref. C-5)
 Circulo de Durensan Vigués. Vigo. C/Venezuela,

48, Entrechan. Tels.; (986) 410 683/422 519. (Ref. C-6) Asociación Manchega de Usuarios de CBM-64 (A.M.U. C-64). C/Blasco de Garay, 10. Albacete.

· Club MICRO-ESPLAI. Gran Via, 682. Pral. (Entre Bruch y Gerona) de Barcelona. (Ref. C-14).

Quisiera saber si existe algún Club Commodore en

Sevilla. Francisco Javier Recio Lamata. C/Fernán Caballero, 22. Sevilla-1. (Ref. C-17).

Agradecería mucho a todo el que pudiera informarme sobre la creación del Club Commodore en Girona, Daniel Bassas, Gerona capital, Plaza Marqués de Camps, 16, 3º, 2.º. Tel. 21 27 26. (Ref. C-18).

• Club Ensenada de Santander. Club Ensenada. C/Sta. Lucia, 1, 1º. Tel.: 22 59 92. (Ref. C-19).

Estoy formando un club de intercambio de programas (en cinta) para VIC-20. Jaime Vidat Forteza.
 C/Sor Damiana, 16-1º S. Francisco (Pil. lari). Palma

de Maliorca. (Ref. C-20).

PARA LLEGAR MAS LEJOS...

SPECTRAVIDED









MICROSOFT BASIC * FORTRAN 80 * Microsoft Logo * COLECO cartuchos CP/M * Cobol 80 * MACROENSAMBLADOR Z80 * Ensamblador CP/M

MSX. Es el sistema escogido per 16 labricantes japoneses de ordenadores (SONY PANASONIC, PIONER, SANYO, SPECTRAVIDEO, alc.) siendo compatibles indos les programas creados per cada uno de los labricantes.

CARACTERISTICAS GENERALES	SPECTRAVIDED 328	SPECTRAVIDEO 318	APPLE II e	COMMODORE 64	ATARI 800 xL	BBC MODELO B	DRAGON 32	ORIC ATMOS
PROCESADOR	280 A	Z80 A	6502	6510	6502	6502	6809 E	6502
RELOJ EN MHZ	4 MHz	4 MHz	1 MHz	1 MHz	1.8 MHz	2 MHz	1 MHz	1. MHz
MEMORIA STANDARD EN RAM	80 K	32 K	84 K	64 K	16 K	32 K	32 K	48 K
AMPLIABLE HASTA	144 K	144 K	-	-	64 K	-	64 K	64 K
MEMORIA STANDARO FILKOM	32 K	32 K	16 K	20 K	24 K	15 K	16 K	16 K
AMPLIABLE HESTA	96 K	96 K	-	40	_	-	w.	-
MICROSOFT BASIC EXTENDIDO	Si	SI	7 -	-	-	NO NO	SI	Si
NUMERO DE TECLAS	87	51	62	66	57	73	53	57
TECLADO CON GRAFICOS PREDEFINIOUS	SI	SI	1/ -	Si	SI	SI	NO	NO.
TECLAS DE FUNCION PROGRAMABLES	10	10		8	4	10	NO	ND
MANDO JOYSTICK INCORPORADO	NO	23	_	NO I	NO NO	NO	NO.	NO
COLORES		16	18	16	16	16	9	B
SPRITES		32	-	8	8	-	2	-
RESOLUCION (Puntos de pantalla)	256=192	256= 192	280 - 192	320 - 200	320 - 192	256×640	256×192	240×200
TEXTO EN PANTALLA	40×24	40×24	40×24	40×25	40×24	40 - 32	32×1B	40=28
CANALES DE SONIDO	3	3	1	3	4	1	3	3
OCTAVAS POR CANAL	8	8	4	9	?	3	5	8
A.D.S.R ENVOLVENTE	SI	SI	NO	SI	NO	SI	ND	SI
CAPACIDAD UNIDAD DE DISCO	256 K	256 K	140 K	170 K	127 K	100 K	2	2
CENTRONICS	SI (occional)	SI (opcione)	Si (pecienal)		Si (opcional)	SI (opcional)	SI	SI
RS232	SI (opcional)	SI (opcional)	SI (opcional)		St (opcional)	\$1 (optional)	14	_
BO COLUMNAS	SI (opcional)	SI (opcional)	Si (opcional)	SI (opcional)	SI (opcional)	-	-	-
PRECIO P.V.P.	76.000	49.900	166.618	79.900	58.500	140.000	67.800	59.500

MSX v MICROSOFT son marcas registradas por MICROSOFT CORPORATION CPM es una marca registrada por DIGITAL RESEARCH INC COLECO VISION es una marca registrada por COLECO INDUSTRIES INC.



DISTRIBUIDOR OF CIAL DE SPECTRAVIDEO INTERNATIONAL LTU PASEO DE LA CASTELLANA, 179 - MADRID-16 TELEFONO (91) 279 31 05 DELEGACIÓN EN CATALUÑA: ACE. S.A. Tarragona, 100 - Barcalona-15, Teléfono (93) 325 10 58

YA DISPONIBLE EN



Y EN TODAS LAS TIENDAS ESPECIALIZADAS